

平成3年度 科学研究費補助金 総合研究（A）

音声対話のモデル化とその機械処理 に関する総合的研究

（課題番号 02305010）

研究成果報告書

平成4年3月

研究代表者

白井克彦

早稲田大学理工学部教授

260185043004

は し が き

本報告書は、平成2年度と3年度の2年間にわたり、文部省科学研究費補助金総合研究（A）「音声対話のモデル化とその機械処理に関する総合的研究」によって行われた研究の成果をまとめたものである。

音声対話は、人間におけるコミュニケーションの最も基本的な様態であり、その適切なモデル化は言語学、哲学、認知科学等の中心的課題である。さらに、音声対話を計算機で実現することは、マン・マシン・インタフェースの高度化に不可欠な要素技術の一つである。本研究では音声対話をその行動科学的側面、並びに言語学的側面からとらえ、情報科学的な観点からモデル化し、最終的には工学的な観点から機械処理を可能にすることを目標として進められた。

研究期間中10回の全体的な会合を開いたが、研究分担者以外の研究者の参加も得て、活発に研究が行われた。

取り上げられた主な研究テーマをあげると、

1. 音声対話における言語モデルの研究
 2. 音声対話システムにおける音声言語表現の生成
 3. 対話音声の理解システムの研究
 4. 音声対話システムにおける対話制御の研究
 5. 音声対話研究に必要とされるデータベース像
- などである。

本研究は、音声対話に関心を持つ音声研究者と自然言語処理研究者が共同して本格的な音声対話研究の出発点を定めようとしたものであるが、その目的は十分に達成されたものとする。

平成4年3月

研究代表者 白井 克彦

研究組織

< 研究代表者 >

白井 克彦 早稲田大学理工学部情報学科 教授 (情報工学)

< 研究分担者 >

板橋 秀一 筑波大学電子情報工学系 教授 (情報工学)
市川 熹 日立製作所中央研究所第6部 主任研究員 (情報工学)
寛 一彦 N T T基礎研究所情報科学研究部 室長 (情報科学)
桐谷 滋 東京大学医学部音声言語医学研究施設 教授 (音声言語科学)
樽松 明 ㈱ A T R自動翻訳電話研究所 社長 (言語音声工学)
小林 哲則 早稲田大学理工学部電気工学科 助教授 (情報工学)
千葉 成美 ㈱日本電子化辞書研究所 主席技師 (情報工学)
土屋 俊 千葉大学文学部 助教授 (言語哲学)
豊田 順一 大阪大学産業科学研究所 教授 (情報科学)
中川 聖一 豊橋技術科学大学情報工学系 教授 (情報工学)
中津井 護 郵政省通通信総合研究所 総合研究官 (音声情報処理)
新美 康永 京都工芸繊維大学工芸学部 教授 (情報科学)
橋田 浩一 新世代コンピュータ技術開発機構 研究員 (認知言語学)
原田 康也 早稲田大学法学部 助教授 (統語理論)
広瀬 啓吉 東京大学工学部電子工学科 助教授 (音声言語学)
藤崎 博也 東京理科大学基礎工学部電子応用工学科 教授 (音声言語科学)
古井 貞熙 N T T・H I研音声情報研究部 室長 (音声情報処理)
牧野 正三 東北大学応用情報科学研究センター 助教授 (情報科学)
溝口理一郎 大阪大学産業科学研究所 教授 (計算機工学)
(以上あいうえお順)

研究経費

平成2年度	3, 4 0 0 千円
平成3年度	3, 9 0 0 千円
計	7, 3 0 0 千円

研究発表

< 論文誌等 >

- 1) K. Shirai, E. Kitagawa and T. Endo: "Optimal Construction of Context Sensitive Quantizer for Phoneme Recognition in Continuous Speech", Proc. 2nd Eurospeech, 405-408 (1991).
- 2) K. Hasida: "Reducing Complexity of Constraint-Based Grammars", in Barwise, J., Gawron, J. M., Plotkin, G., and Tutiya, S (eds.), Situation Theory and Its Applications, II, CSLI Lecture Notes Number 26, 405-423 (1991).
- 3) K. Hasida: "Common Heuristics for Parsing, Generation, and Whatever", Workshop on Reversible Grammar in Natural Language Processing, Berkeley, 81-90 (1991).
- 4) K. Hasida: "Dynamics of Symbol Systems: An Integrated Architecture of Cognition," to appear in FGCS '92.
- 5) Y. Yamashita, N. Mizutani and R. Mizoguchi: "Concept Description for Synthetic Speech Output System", Proc. of the ESCA Workshop on Speech Synthesis, AuTRANS (France), 241-244 (1990).
- 6) T. Yamamoto, Y. Ohta, Y. Yamashita and R. Mizoguchi: "Dialog Management System MASCOTS in Speech Understanding System", Proc. of ICSLP '90, Kobe, 1301-1304 (1990).
- 7) T. Yamamoto, Y. Ohta, Y. Yamashita, O. Kakusho and R. Mizoguchi: "MASCOTS: Dialog Management System for Speech Understanding System", Trans. IEICE, E74, 7, 1881-1888 (1991).
- 8) M. Nakatsui and R. Tanaka: "Half-syllabic units for speech processing — An automatic segmentation," J. Acoust. Soc. Jpn. (E), 11, 4 (1990).
- 9) A. Komatsu, E. Ohhira and A. Ichikawa: "Spontaneous Speech Understanding Based on Cooperative Problem-Solving", Trans. IEICE, E74, 7 (1991).
- 10) 大平, 小松, 阿部, 市川: "情報検索における柔軟な対話制御方式", 電子通信学会論文誌 (発表予定) .
- 11) S. Makino, A. Ito, M. Endo and K. Kido: "A Japanese Text Dictation System Based on Phoneme Recognition and a Dependency Grammar", Trans. IEICE, E74, 7, 1773-1782 (1991).
- 12) S. Matsunaga, S. Sagayama, S. Homma and S. Furui: "A Continuous Speech Recognition System Based on a Two-Level Grammar Approach", Proc. of

ICASSP'90, 589-592 (1990).

- 13) S. Furui: "Speech Recognition Using VQ-Codebooks Representing Hierarchical Spectral Dynamics", Trans. IEICE Jpn., J74-A, 5, 750-757 (1991).
- 14) S. Matsunaga, S. Homma, S. Sagayama and S. Furui: "Continuous Speech Recognition Using a Dependency Grammar and Phoneme-Based HMMs", Trans. IEICE, E74, 7, 1826-1833 (1991).
- 15) S. Furui: "Recent Advances in Speech Recognition", Proc. Eurospeech 93-10 (1991).
- 16) 山本, 中川: "多重様相論理 TMS", 人工知能学会誌, 6, 3 (1991).
- 17) A. Kurematsu: "Perspective of Telephone Interpretation Research", Pacific Rim International Conference on Artificial Intelligence (1990).
- 18) A. Kurematsu, H. Iida, T. Morimoto: "Language Processing in Connection with Speech Translation", Speech Communication, 10, 1 (1991).
- 19) 樽松: "自動翻訳電話のための音声情報処理", 人工知能学会誌, 6, 3 (1991).
- 20) T. Morimoto, K. Shikano, K. Kogure, H. Iida, A. Kurematsu: "Integration of Speech Recognition and Language Processing in a Japanese English to Japanese Spoken Language Translation System", Trans. IEICE, E74, 7 (1991).
- 21) A. Kurematsu: "Overview of Speech Translation for the Automatic Telephone Interpretation", International Joint Conference on Artificial Intelligence (1991).
- 22) Y. Niimi, Y. Kobayashi: "A Top-down Discourse Analysis in a Speech Dialogue System", Proc. of European Signal Processing Conference, 1275-1278 (1990).
- 23) Y. Kobayashi, M. Omote, H. Endo, Y. Niimi: "SUSKIT-II ---- A Speech Understanding System Based on Robust Phone Spotting", IEICE Trans., E74, 7, 1863-1869 (1991).
- 24) Y. Niimi, Y. Kobayashi: "A Speech Interface to an Information Retrieval System", Studia Phonologica, XXIV, 96-110 (1990).
- 25) K. Takagi, S. Itahashi: "Formant Frequency extraction by moment calculation of speech spectrum", J. Acoust. Soc. Jpn., E12, 1, 47- (1991).
- 26) S. Itahashi: "Creating Speech Corpora for Speech Science and Technology", Trans. IEICE, E74, 7, pp. 1906-1910 (1991).
- 27) 高橋, 板橋: "ニューラルネットワークを用いた日本語解析の試み", 情報処理学会論文誌 32, 10, 1330-1337 (1991).

< 口頭発表 >

- 1) 白井: "音声対話システムにおける音響処理と言語処理の接点", 人工知能学会 A I シンポジウム, 人工知能学会研究会資料 SIG-F/H/K-9101-1, 1-18 (1991).
- 2) 渡戸, 遠藤, 小林, 白井: "事例に基づく推論による音声認識手法の検討", 日本音響学会秋季講演論文集, 2-P-24 (1991).
- 3) 吉田, 野村, 山下, 溝口: "対話管理におけるMOPを用いたユーザ発話の予測", 日本音響学会秋季講演論文集, 2-P-22, 181-182 (1991).
- 4) 柴田, 山下, 溝口: "合成音声出力における対話管理について", 信学技報, SP91-109, 9-16 (1992).
- 5) 浅野, 広瀬, 藤崎: "音声応答システムにおける対話管理手法", 日本音響学会秋季全国大会講演論文集, 3-P-17, 325-326 (1991).
- 6) 浅野, 広瀬: "相互主導型対話機能を持つ音声応答システムとその対話管理手法", 電子情報通信学会技術研究報告 (音声研究会), SP91-110, 17-24 (1992).
- 7) 浅野, 広瀬, 藤崎: "音声応答システムにおける対話管理部の実現", 日本音響学会春季全国大会講演論文集, (発表予定).
- 8) 山本, 小林, 中川: "対話における曖昧さを解消するための質問生成システム", 第5回人工知能学会全国大会 (1991).
- 9) 中川, 山本, 田口, 小林: "音声による対話システムにおける言語処理", 1991年度電気関係学会東海支部連合大会 (1991).
- 10) 山本, 小林, 中川: "音声対話文における助詞落ち・倒置の分析とその解析手法", 自然言語処理の新しい応用シンポジウム (1992).
- 11) 新美: "音声言語からみた対話処理の課題", 人工知能学会研究会資料, SIG-F/H/K-9101, 31-32 (1991).
- 12) 福田, 板橋: "係り受け結合頻度を用いた複合名詞解析の一方法", 情報処理学会全国体会 (42回), 1C-5, 3-9~3-10 (1991).

会 議 録

第 1 回会合

日時：1990年7月14日（土） 午後1時より

講演：1)土屋俊（千葉大）「言語行為論からみた音声対話のモデル」
2)新美康永（京都工繊大）「音声対話の機械処理の課題」

第 2 回会合

日時：1990年9月29日（土） 午後1時より

講演：1)島津明（NTT）「対話理解研究の課題－NTT基礎研究所の取り組み」
2)江原暉将（ATR）「ATR対話データベース」

第 3 回会合 「シンポジウム」

日時：1990年11月17日（土）

シンポジウム

「Workshop on Statistical Approaches to Spoken Language Processing」
（ICSLP90のサテライトシンポジウム。学術振興会第512委員会、
総合研究A「マルコフモデル・ニューラルネットを包含する新しい音声
認識手法の総合的研究」（代表：中川聖一教授）共催）

第 4 回会合

日時：1990年12月8日（土） 午後1時30分より

講演：1)橋田浩一（ICOT）「記号制約のポテンシャルエネルギー」
2)伊藤昭，樋口英幸（通信総研）「対話による計算機利用支援」

第 5 回会合

日時：1991年1月11日（金） 午後3時より

講演：1)溝口理一郎（阪大）「対話管理システムについて」
2)山本幹雄，中川聖一（豊橋技科大）「対話参加者の信念の表現と推

第 6 回会合

日時：1991年3月30日（土） 午後1時半より

講演：1)山下洋一（阪大）「概念表現を利用した汎用音声出力インタフェース」
2)浅野康治，広瀬啓吉（東大）「知識表現からの文章合成システム

第7回会合

日時：1991年5月25日（土） 午前11時より

講演：1)千葉成美（EDR）「EDR電子化辞書について」

2)田中穂積（東工大）「YAGLR：一般化LR法による新しい構文解析アルゴリズムとその評価」

第8回会合 「夏の研究会」

日時：1991年8月1～3日

セッション1S

1)馬塚れい子（Duke大）「日本語の文理解について－統語処理を中心に－」

セッション1A 対話における言語現象

1)竹沢寿幸（ATR）「ATR対話データベースに見られる言語現象の分析と効率的な音声認識用の文法について」

2)寛一彦（NTT）「実音声対話データからの示唆」

3)小林哲則（早大）「不要語の分類－不要語は不要か？－」

4)山本幹雄（豊技大）「音声対話におけるいくつかの言語的な特徴について」

セッション1N

1)加藤恒昭（NTT）「自然言語処理における対話理解」

2)藤崎博也（理科大）

セッション2A 音声・言語処理

1)橋田浩一（ICOT）「制約の力学」

2)岡田美智男（NTT）「音声言語システム(Spoken Language System)について」

3)河原達也（京大）「Dempster-Shafer 理論に基づく多レベル知識の統合に基づく音声理解」

4)伊藤彰則，牧野正三（東北大）「日本語文節オートマトン」

5)中津井護（通総研）「電気通信フロンティア研究計画の概要紹介」

セッション2B 対話処理1

1)中島信弥（NTT）「協調的対話における韻律情報の役割について」

2)広瀬啓吉（東大），藤崎博也（理科大）「音声対話システムにおける音声合成について」

3)新美康永（工繊大）「対話音声中の不要語のスポティング」

4)鈴木浩之（松下）「日本語発話を逐次的に解釈する」

セッション2C 対話処理2

1)山下洋一（阪大）「対話音声認識におけるユーザ発話の予測」

2)飯田仁（ATR）「言語運用の知識と次発話タイプの予測」

3)大塚裕子(計量計画研)「国語学からみた対話音声研究について」

セッション2N

1)土屋俊(千葉大)「なぜ音声インタフェースは役に立ちそうもないか」

セッション3A データベース

1)板橋秀一(筑波大)「音響学会連続音声データベースについて」

2)速水悟(電総研)「案内タスクの連続音声DBのための対話テキスト」

3)桑原尚夫(西東京科大)「音声合成用データベース」

4)鹿野清宏(NTT)「DARPAのATISデータベースについて」

5)白井克彦(早大)「対話音声データベースにむけて」

第9回会合

日時：1991年12月4日 3:00より

講演：1)片桐恭弘(NTT)「談話理解と状況意味論」

2)市川熹, 天野明雄, 北原義典(日立)「口語調音声による情報検索システムの試み」

3)高橋直人, 板橋秀一(筑波大)「相互結合型ニューラルネットによる日本語の係り受け解析」

第10回会合 「成果報告会」

日時：1992年2月8日(土) 11:00より

講演：1)白井克彦(早大)総括

2)菊池英明, 小林哲則, 白井克彦(早大)「自然な模擬対話を収録するために」

3)中川聖一, 山本幹雄, 小林聡, 田口勝豊(豊橋技科大)「音声対話のいくつかの言語的な特徴と対話システム」

4)山下洋一, 溝口理一郎(阪大)「機械との音声対話におけるユーザ発話の予測について」

目次

音声対話のモデル化について	白井克彦 (早大・理工)	1
音声言語処理からみた対話処理の課題	新美康永 (京都工繊大)	7
対話音声データベースの課題	板橋秀一 (筑波大・情報)	9
自動翻訳電話の動向と課題	樽松明 (ATR自動翻訳電話研究所)	18
制約の力学	橋田浩一 (ICOT)	32
音声入出力のための対話管理について	山下洋一, 溝口理一郎 (阪大・産研)	38
機械との音声対話における次発話の予測について	山下洋一, 溝口理一郎 (阪大・産研)	48
日本語音声応答システムー特に対話管理についてー	広瀬啓吉, 浅野康治 (東大・工)	52
音声対話のいくつかの言語的な特徴と対話システム	中川聖一, 山本幹雄, 小林聡, 田口勝豊 (豊橋技科大)	61
音声対話における不要語・復唱・倒置および助詞の役割について	小林哲則, 白井克彦 (早大・理工)	73
日本語文音声認識のための統語処理アルゴリズムー機能語予測CYK法ー	伊藤彰則, 牧野正三 (東北大・応情研)	79
音声インターフェースを持つ情報検索システムと談話処理	小林豊, 田辺雅則, 新美康永 (京都工繊大)	88
口語調音声による情報検索システムの試み	市川熹, 天野明雄, 北原義典 (日立・中研)	100
自然な模擬対話の収録のために	菊池英明, 小林哲則, 白井克彦 (早大・理工)	109
ATR対話コーパスを利用した音声認識のための構文規則	保坂順子, 竹沢寿幸, 江原暉将 (ATR自動翻訳電話研究所)	119
音声言語のパーキングについて	岡田美智男 (NTT・基礎研)	133
A Study on the Roles of Prosody in the Cooperative Dialogue ... Shin'ya Nakajima(NTT HI Lab.), J.F.Allen(Univ. of Rochester)		143
発表論文別刷		153

音声対話のモデル化について

白井克彦

(早稲田大学・理工学部)

1. はじめに

音声認識および音声合成技術の応用上で最も大きな目的は、人間－機械間の自然な会話を可能にすることである。しかしながら、音声言語に関する機構や知識を機械上に作成することは、人間の思考や記憶のメカニズムを何らかの限定を伴うとしても実現することであるから、当然極めて困難な問題である。

従来、音声認識の研究において、音声の音響的特徴と識別に関して多くの研究努力が注がれてきた。その結果、かなりの高精度化も実現されてきたのであるが、これを発展させて音声対話システムを作る場合、音韻認識部に言語処理を組み合わせることが試みられてきた。この場合の言語処理は、書き言葉に対する自然言語処理を指すが、近年は機械翻訳などを中心にして発達してきたものである。音声処理と自然言語処理の両技術を融合して、音声言語の理解と生成システムを作ることによって、音声対話システムを構成する研究は、近年、多く試みられるようになってきたが、まだ本格的なシステムの構築には至っていない。この理由は、第一に音声認識によって得られる認識結果というものは、人間の発声上のゆらぎや個人差、音波の伝送過程における雑音や歪の混入などによって、それを記号列に表した場合多くの曖昧性を含んでいるために、自然言語処理部分では、入力手段に誤りの少ない文字言語に対すると同様な処理文法だけでは対応が困難な点である。第二は、より本質的であるが、自然な音声対話は従来の音声認識と自然言語処理の対象としてきたものとはかなり内容的に異なるものであるということである。これまでの音声認識は、単語認識に始まり連続音声認識に至っているが、認識できる対象はかなり文法的に制約のある文章を明確に発声するという条件で、日常的な自由な話し方は一般に許されない。また文字化した言語表現で自然言語処理上問題とされてきた複雑な意味解析などの諸問題は、当然、話し言葉の中でも存在しているが、音声対話では文法からはずれた表現や省略、不完全な表現など従来の自然言語処理におけるものとは趣きの異なる問題が多く現れている。

したがって、ここで問題となるのは実際の音声対話について、音声認識と自然言語処理の立場で何が重要であるかという点と音声対話全体で新しい問題は何かということである。

本研究では、音声研究、自然言語研究、認識科学研究、知識処理研究など、参加した研究者のそれぞれの立場で研究を進め議論を行った。音声対話の研究は、広範な内容を含みこの研究期間で問題が充分整理されたとはいえないが、研究の

緒に着いたとは考えられる。音声対話について本格的な研究を推進するには、まず適切なデータが必要である。これまで、本研究のような目的で対話音声データをまとめた例は、ATR対話音声データベースを除けば大変少ない。本研究グループとは別の組織としてであるが対話音声データベース委員会（代表 板橋秀一）と密接な関係を持って活動した。そこでは1年目の議論の結果は、現在の音声認識の技術である程度は扱い得るような明瞭で文法も極端に逸脱していない文章音声データベースを集めて研究を進めるべきであるという意見が大勢であった。そこで集められた対話音声データベースは共通研究課題として提供されることとなった。2年度は、データベースにおいては共通データベースおよび個々のグループの取りあげているタスクで研究が進められている。

2. 音声対話に見られる特徴

話し言葉には、省略、倒置、言い直し、中断など、通常書き言葉を対象としている文法には合致しない表現が数多く現れる。表1にいくつか例をあげたが、たとえば省略について考えると、省略によって意味が不明になったり、曖昧になったりする場合は少なく、むしろ簡略化された表現によって、意味が明快に伝わる場合が多い。

その他の例においても、まずその表現を前後の文脈も考慮に入れたとして、文法的、内容表現的に正しい文章にすることが可能であるかどうかを問題にするべきである。

次に、その表現が少なくとも話者にとって自然に用いられる理由が調べられる。たとえば自己調整を考えると、表現内容の話者の思考過程に関係しているときと聴き手の正しい理解が困難になることを配慮して生ずる場合があげられるが、いずれも適切な表現を求めてなされる発話現象といえる。当然ながら、あいづちや不要語なども聴き手の了解性を高める作用が大きい。

こうして考えると、音声対話システムでは書き言葉で使われているような整った文体ではなく、音声対話規則あるいは音声対話モデルに基づく解釈や表現法が必要となることが明らかである。近年の発話行為に関する研究や対話に関するいくつかの研究は、この方向の重要な内容を含んでいるが、機械処理に適う体系を作ることが課題である。

3. 音声対話における情報の流れ

図1のような2人の対話者(A, B)における情報交換を音声認識の立場から見てみる。話者Aが文音声を話す過程には図2(a)のような内容がある。発話内容が文章として表現されれば、これを音声に変換する過程は、従来から音声合成技術として研究されているが、発話意図I(A)がいかに組み立てられ、文章の形に変

換されるかは、音声対話システムの中の重要な要素であり、大きな問題である。発話意図の決定はその時点における対話者Bの発話内容M(B)と観測される外界の状況、さらにAの持つ知識ベースとこれまでの談話構造や心的情報などによってなされる。その決定過程には、発話意図や対話のゴールが存在し、様々なプラン・スキーマに従って談話が組み立てられる場合もあるが、相手の発話内容に応じて反射的に対応するような、反射的発話と常識や比喩などに基づく連想的応答なども実際の発話に多くの割合で存在している。

図2(b)は話者Aの受聴過程を示す。音声認識における問題は、処理の各過程における情報内容を考えると、図3(a), (b)のように様々な包含関係が生ずることである。極端な場合、図3(b)のように、V(B)すなわちBの発話の音声波に含まれる情報とそれをAが受聴して理解した結果、Bが意図したであろうと考える意味内容M(B)はV(B)とはかなりかけ離れたものになることもある。もっとも、同様のことは、図2(a)の発話プロセスでも生じ得ることで、本当の発話意図I(A)と生成された文U(A)の適切性、妥当性には問題がないわけではない。

結局、発話者Aの発話意図I(A)とこれを受ける対話者bの理解した意味内容M(B)の間には、図4(a), (b), (c)等の様々な状態が存在する。図4(a)はI(A)とM(B)が全く異なっている場合で、対話に大きな困難が予測される。図4(b)では、音声を通じた健全な通信が行われている。図4(c)では、音声を通じていない共通理解部分が生じている。A, Bに固有な常識や共通世界知識によるか、この対話において起こった誤解に基づく共通認識とでも呼ぶべき部分が存在する。

4. まとめ

音声対話の全体モデルを考えると、まず対話者が2人以上存在すること、話題となる外界の状況があるので、状態として少なくとも、2人の心的状態、談話の状態、外界の状態の4種類が存在していて、それらの間に情報が流れることになるが、一般にはこれに物理的なアクションが加わる。次に、情報が音声表現を取るために、情報形態が数回変化することとその過程でその情報内容が変化する可能性があることである。

したがって、音声対話システムの問題は、4種類の状態と知識ベースから情報表現がどのように生成されるかという点と音声理解の問題となるが、この中に多くの曖昧性や冗長性が存在していることが重要である。つまり、完全な整合性を持って全体のシステムが動作することは、あまり期待できず、対話者の満足度は、対話を抵抗感なく自然に続行できて、目標があれば、それが達成されることである。

現象名	例
言い換え (同内容)	ディズニーランド、東京ディズニーランドです。
言い直し (構文的变化)	晴海通りからまっすぐ行って、 <u>行けば</u> 有明に着きます。
言い差し (中断)	A : 靖国通りを <u>通</u> って、皇居の周りを走って、 B : <u>晴海通り</u> 。 A : 銀座を通り抜けて晴海通りに出ます。
あいづち	A : 晴海通りから行く場合にはですね、 B : <u>そうですね</u> 。
自己調整	早稲田通りで銀座方面まで行きたいので、外苑東まで言 って、 <u>それから、えっと、じゃあ、</u> とりあえず銀座まで 行きます。
不要語	<u>えっと</u> 、横浜までいきたいんだけど。
照応	A : 玉川インターから第3京浜に乗って下さい。 B : <u>そこは</u> 混んでいませんか？
倒置	首都高速の方が早いよね、早さとしては。
省略	これで行くと、横浜（に着くのは）何時ですか。
文の中断	明治通りは混むから他の方法がいいのだけれど。
助詞落ち	山手通り（を）使う手もあるよね。
副用言	やっぱり、もう、まだ、しか
接続詞	しかし、だけど、でも
接尾辞	とか、ぐらい
終助詞	よ、ね、よね

表1 音声会話に生ずる表現の例

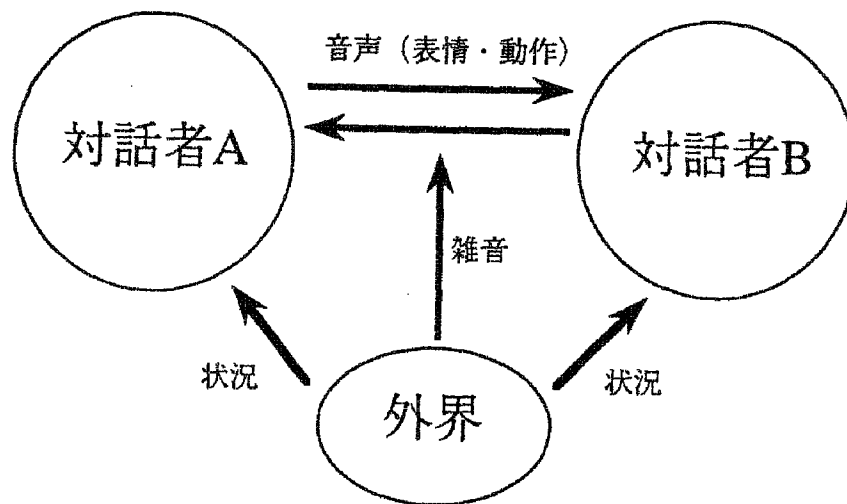


図1 対話の状況

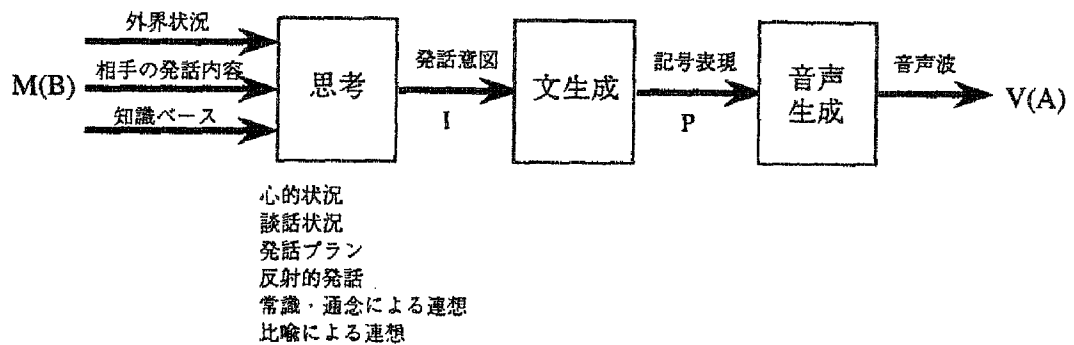


図2(a) 話者Aの思考と発話

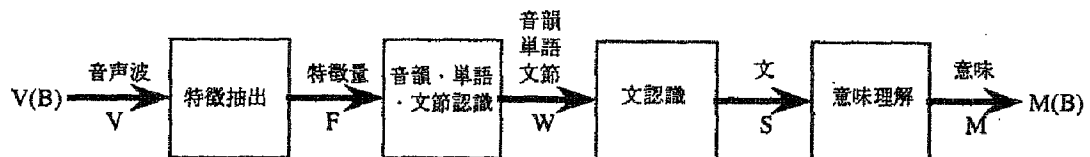


図2(b) 話者Aの受聴

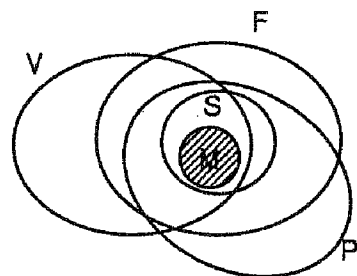


図 3 (a)

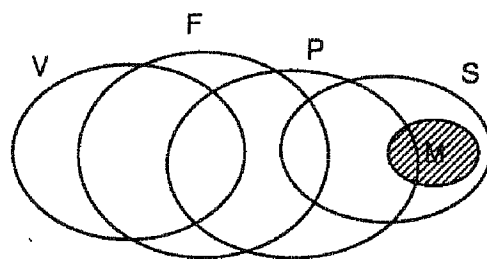


図 3 (b)

図 3 聴き手の理解過程の情報内容

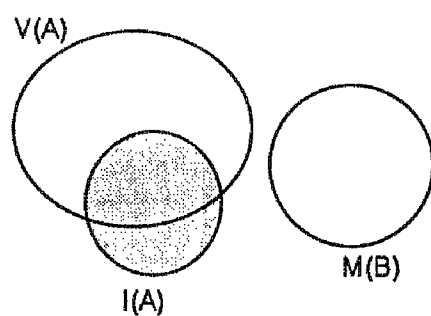


図 4 (a)

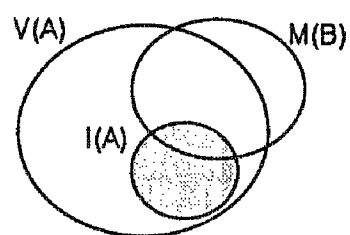


図 4 (b)

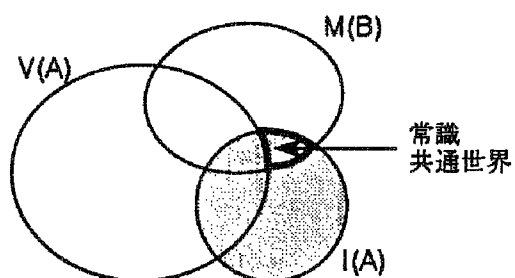


図 4 (c)

図 4 対話者(A,B)の間の持つ情報内容の関係

音声言語からみた対話処理の課題

新美 康永
京都工芸繊維大学

1. まえがき

音声言語による対話のチャネル構造を、OSIの7階層モデルにならって示すと図1のようになろう。送受信者間で伝達したい情報が、自然言語という符号系によってコード化されたのち、音波という物理量によって伝達される様子を示したものである。さらに対話の目的によって決まるプロトコルと「情報」とは何かを決める「話者モデル」が上位の層として付されている。ここでは、このような構造をしたチャネル

話者モデル層
応用層
情報層
言語層
音波層

図1 音声チャネルの構造

を「音声チャネル」と呼ぼう。音声チャネルを用いた対話の形態には、(1)対面对話、(2)テレビ電話、(3)通常の電話、(4)機械との対話、等が考えられる。音声対話の本質を考えるためには、このような広い立場から考察を加える必要があるが、ここでは(4)の人間と機械との対話を実現するために、考えなければならない課題のいくつかを提起しよう。

2. 音声対話の機械処理の課題

音声対話の機械処理の課題は、標語的にいえば、(1)人間が機械と対話する、(2)機械による音声認識は不完全、(3)対話として自然な合成音声、の3点に要約されよう。

2.1 機械と話するときの人間のモデル

音声チャネルは人間と人間とが意思の伝達をはかるために、人類の発達と同時進行的に発達してきたコミュニケーション・チャネルである。単語認識装置の実用化による極めて少数の例を除いては、人類がこれを他の用途に使用する経験はほとんどなかった。したがって、人間が機械と対話する場合、どのように振舞うかはまったく未知といってよい。

現在、(1)人間-人間の対話、および(2)人間-模擬システムの対話を調査することによって、人間の行動様式が調べられている。(1)は人間が機械と”しゃべる”とき、人間に対するのと同じ振舞いをするか否かは不明であるのであまり直接的ではないが、人間と人間との対話を多数集めて、これを分析することは有益であろう。(2)の模擬システムとは、音声認識システムと問題解決器は人間であって、入力に故意に雑音を付加して聴き取り誤りを誘発したり、出力側には分析合成型のボコーダを挿入して機械らしさを演出したシステムである。(3)の実現が望めない現在では、(1)の場合よりは有益なデータが期待できそうな方法であり、いくつかの研究機関で実施されている。しかし、雑音の付加による人間の聴き誤りの傾向と音声認識システムの誤りの傾向とは、まったく異なっていること、人間は広範囲の知識を利用して、聴き誤った部分や聴き取れなかった部分を補っているので、この方法によっても十分正確なデータが得られるとは限らない。実時間で動作する性能のよい音声認識システムの実現が強く望まれる。

2.2 対話における発話の特性

人間－機械の対話の例とするにはあまり適切ではないが、人間－人間の対話に現われる発話の特性を、音声理解システムなどが対象としている朗読文に近いものと比較してみよう。1. で定義した音声チャンネルの階層と対比させると次のようになる。

- ・物理（音波）層――人間と人間の間の自発的な発話の大部分がそうであるが、発話の「なまけ」が著しい。特に文末、語尾、閉鎖音の直前などで、声帯の緊張が緩み、言語音としての特性がくずれてくる。このような部分を通常の分析法で分析識別しようとしても、雑音的な結果しか生じない。分析の早い段階で、発声に緊張を欠いた部分を検出する技術の開発が必要である。

- ・言語層――書き言葉の文法からは逸脱した種々の現象が現われる。例えば、文の中断、非文法的な文の中止、言い直し、無意味語句の挿入、同一語句の繰り返し等。対話中の発話の解析のためには、このような種々の不規則性を能率よく扱うことのできる解析法の開発が必要となる。

- ・情報レベル以上の層――対話の文脈に依存した語句の省略や代名詞の使用、韻律（声の高低、強弱、緩急）を利用した強調が起こる。また対話の構造を決める有力な手懸かりとなるいわゆる cue word の挿入等がある。自然言語処理において文脈解析と呼ばれている部分が担当する領域であるが、このレベルの現象と音響信号（とくに韻律信号）とが直接結びつくことが多いので、音声認識・合成の両面から総合的に研究を進めることが必要である。

2.3 対話制御における課題

人間と機械の間でどのような情報のやりとりができれば、人間にとって満足のいく対話と言えるのだろうか。図1のモデルでいえば、主として応用層、話者モデル層に属する問題である。一般的な対話制御の問題に深入りすることは避けて、ここでは”音声認識システムは不完全である”ことによって生ずる対話制御の問題について考えてみよう。このような条件での対話制御の最下層（単一の情報の授受を制御する場合）では、最適性（最短時間で最大限の情報を獲得する）、安定性（人間からの回答に対する予測可能性が最大に、回答の誤り易さが最小になるように質問を構成する）、適応性（音声認識システムの認識性能に応じた質問／回答の動的構成能力）などの機能が必要である。さらに認識結果の確認を、いつ、どのような方法でさりげなく行うかも重要な心理的要因であろう。

2.4 自然な合成対話音声

人間－機械対話系における機械側の合成出力が自然であるためには、朗読文として自然に聞こえる合成音ができるだけでは不十分であろう。3.2 で考察したような文脈に依存した種々の現象を十分反映した、テキストの生成や韻律の付加が不可欠であろう。さらに、利用者に心理的な負担（例えば、機械側の出力が終わったことを確認するのに気を使うといったこと）をなるべく与えないようにする配慮も必要であろう。

3. あとがき

音声をメディアとするコミュニケーション・チャンネルは、しばしば多のメディアと共存して使用される。しかし、本稿では音声チャンネルが単独で使用され、しかも対話のパートナーが機械である場合の種々の問題を提起した。

対話音声データベースの課題

板橋 秀一

(筑波大学 電子・情報工学系)

1. 音声データベース

音声分析や自動認識・合成の研究・開発において音声データベースが必要であることはこの10年間で十分認識されるようになった。現在、日本で公開されているものは電子協、ATR、電総研、東北大・松下、それに文部省科研費によるもの等があるが、内容・普及度とも、米国の状況に比べると遅れを感じる。内容的に整っているものはATRのデータベースであるが、話者数が少ないことと有料であるため、普及に時間がかかるように思われる。日本の大部分の音声研究者が加入できるような組織を構成し、その中では殆ど無償で流通できるような機構が望まれる。大学・官庁・企業を包含する組織が必要である（例えば学会、協会）。

音声データベースは音声情報処理用と、音響・音声学用に大別できる。後者は文部省プロジェクトで進行中なので、ここでは前者に話題を絞る。研究・開発用としてはラベルがついていることが望ましいが、認識システムの性能評価用ならば、ラベルは特に必要がないものと考えられる。またDATインターフェースを利用することにより、作成・利用面での簡素化が図れる。

今後必要な検討事項としては、次のようなものが考えられる。

- 1) タスク：文献検索等
- 2) 発声様式
 - a) 目標付き自然対話
 - b) 読み上げ文
- 3) 規模：延べ10,000文
 - 文の種類：1,000
 - 1人当たり：100文
 - 語彙数：1,000語
 - 話者数：100～1,000人
 - 発声回数：1～2回
- 4) 実環境下：騒音、(電話回線)
- 5) 方言：共通語、東北、関西、九州等
- 6) 記録媒体 DAT/CD-ROM/カートリッジテープ
- 7) ソフトウェアの公開
 - a) AD/DA・編集プログラム (WS用、PC用、DAT-IFも考慮)
 - b) 半自動ラベリングプログラム
 - c) 標本化周期変換プログラム

- d) 自動スコアリングプログラム
- e) 視察ラベリングのマニュアル
- 8) 音声合成用を含むテキストデータベース
- 9) (連続) 音声テキストの選定基準・方法の確立
 - a) 音素・音節出現頻度, digram, trigram (情報エントロピー)
 - b) 構文・意味・pragmaticsの考慮 (perplexity等)
 - c) Bigram, trigram : 日本語では単語単位のものでは有用か? 音節単位か?
- 10) 長期計画を立て、毎年少しずつでも良いから積み上げ方式で作成する。
作業・費用の分散も可能—複数組織による分業も考えられる。
データ収録とラベルづけ作業の分離等。
- 11) 標本化周波数
8, 10, 12, 16, 20, 24, 32, 48, 44.1 KHz
- 12) 研究開発用と評価用のデータの分離
- 13) 音声の特性を生かしたDBMS
- 14) 半自動音声データ収録システム

文章サンプル選定の考え方

- 1) 助詞・助動詞等の付属語の大部分を含むこと。
- 2) 名詞以外の自立語の相当部分を含むこと。
- 3) 単文・重文・複文を適度に含むこと。
- 4) アクセント型・最小音素対→単語サンプルで考慮すれば良いか?

2. 対話音声研究関連文献リスト抄

対話音声研究の歴史は比較的新しいが、関連論文は既に多数報告されている。そこで筆者の手許にある幾つかの文献から、関連するものを抜き出してみた。大部分は表題を参照し、それで不明な場合は論文自体も参照するという方法をとったため、必ずしも密接な関連があるものばかりとは限らない。また表題に「対話」に関連した用語が入っていないものは落ちている。文献は以下の通りである。

- 1) 日本音響学会講演論文集(1985～1991)
- 2) 日本音響学会誌(1984～1991)
- 3) 電子情報通信学会論文誌(1986～1991)
- 4) Proc. ICSLP90 (1990)
- 5) Proc. ICASSP86(1986)
- 6) Proc. European Conference on Speech Technology 87(1987)
- 7) Proc. Eurospeech89(1989)
- 8) Proc. Eurospeech91(1991)

音声対話文献リストその1

1. 音声入力による情報検索システムにおける口語文受理文法の検討
北原義典, 天野明雄, 村山秀樹, 市川:
日本音響学会平成3年度春季研究発表会講演論文集, 3-5-7, pp.101-102
2. 非接触型話者照合システムのヒューマンインターフェース方式の検討
小倉泰憲, 橋本新一郎:
日本音響学会平成3年度春季研究発表会講演論文集, 3-6-12, pp.133-134
3. 音声認識装置のビデオ予約への応用
大蜘蛛篤, 野村和也, 桑野裕康, 平岡省二, 渡辺泰助, 野村泰宏, 周田悦治:
日本音響学会平成3年度春季研究発表会講演論文集, 3-6-16, pp.141-142
4. 知識表現からの音声合成システムにおける対話管理部の役割と表層文の生成
浅野康治, 広瀬啓吉, 藤崎博也, 服部雅一:
日本音響学会平成3年度春季研究発表会講演論文集, 1-6-19, pp.243-244
5. 概念からの音声合成における入力表現の抽象度レベルに関して
山下洋一, 柴田宜宏, 溝口理一郎, 水谷直樹:
日本音響学会平成3年度春季研究発表会講演論文集, 1-6-20, pp.245-246
6. 騒音環境下でのワードスポッティングによる音声認識における不要語の影響
金澤博史, 坪井宏之, 竹林洋一:
日本音響学会平成2年度秋季研究発表会, 2-8-8, pp.61-62
7. 部分マッチングを考慮したワードスポッティング音声認識の検討
北井幹雄, 今村明弘:
日本音響学会平成2年度秋季研究発表会講演論文集, 2-8-9, pp.63-64
8. プランモデルと発話タイプに基づく音声理解のための対話処理
坂井伸輔, 野口淳, 畑崎香一郎:
日本音響学会平成2年度秋季研究発表会講演論文集, 3-8-18, pp.129-130
9. 知識表現からの文章音声合成システム
藤崎博也, 広瀬啓一, 浅野康治:
日本音響学会平成2年度秋季研究発表会講演論文集, 2-6-6, pp.231-232
10. 合成音出力のための概念表現と対話管理
山下洋一, 溝口理一郎, 水谷直樹:
日本音響学会平成2年度秋季研究発表会講演論文集, 2-6-7, pp.233-234

11. 構文的曖昧性を含む会話音声文に対する意味処理方式の検討
永井明人, 石川泰, 中島邦男:
日本音響学会平成 2 年度春季研究発表会講演論文集, 3-3-15, pp.99-100
12. 音声理解システム SPURT-I のための対話管理機構
山本哲也, 尾崎弘, 堀雅洋, 溝口理一郎:
日本音響学会平成元年度秋季研究発表会講演論文集, 3-1-13, pp.101-102
13. 音声応答システムにおける使い勝手の良さについての考察
片岸一起, 谷戸文広, 樋口宜男:
日本音響学会平成元年度秋季研究発表会講演論文集, 3-4-14, pp.127-128
14. 音声認識システムにおける massively parallel の可能性について
速水悟, 田中和也, 太田耕三:
日本音響学会昭和 63 年度秋季研究発表会講演論文集, 1-3-21, pp.41-42
15. 会話文における文節構造のモデル化について
岡田美智男:
日本音響学会昭和 63 年度秋季研究発表会講演論文集, 3-3-16, pp.99-100
16. ペトリネットを用いた実会話のやりとりのモデル
小坂直敏:
日本音響学会昭和 63 年度秋季研究発表会講演論文集, 3-3-17, pp.101-102
17. 内線電話番号を対象とした音声 Q・A システムの構成
片岸一起, 谷戸文広, 樋口宜男:
日本音響学会昭和 63 年度春季研究発表会講演論文集, 3-2-1, pp.95-96
18. 対話文音声認識における単語候補予測及び絞り込みの効果
野村和也, 星見昌克, 二矢田勝行:
日本音響学会昭和 63 年度春季研究発表会講演論文集, 3-2-6, pp.105-106
19. 連続音声認識のための会話音声の特性解析
本間茂, 中津良平:
日本音響学会昭和 62 年度春季研究発表会講演論文集, 3-5-8, pp.105-106
20. 音声理解システムの開発 (2)- 音響処理部 SPREX における支援環境の整備とその評価 -
辻野克彦, 溝口理一郎, 角所収:
日本音響学会昭和 62 年度春季研究発表会講演論文集, 3-5-14, pp.117-118
21. 音声理解システムの開発 (2)- 連想パーサ ASP による言語処理について -
堀雅洋, 溝口理一郎, 角所収, 大隅信, 尾崎弘:
日本音響学会昭和 62 年度春季研究発表会講演論文集, 3-5-15, pp.119-120

22. ϵ -node を有するネットワークモデルに基づく音声会話システム - 到達経路に応じた ϵ -node の重み付け -

小林哲則:

日本音響学会昭和 60 年度秋季研究発表会講演論文集, 2-4-16, pp.81-82

23. ネットワークモデルによる会話音声認識における焦点の表現法

小林哲則, 白井克彦:

日本音響学会昭和 60 年度春季研究発表会講演論文集, 1-4-17, pp.33-34

24. 韻律情報を利用した構文推定およびワードスポットによる会話音声理解方式

小松昭男, 大平栄二, 市川:

電子情報通信学会論文誌 (D), J71-D, 7, pp.1218-1228 (1988)

25. 音声理解システムのための連想パーサ ASP- 認知科学的検討に基づくフレームワークの設計

堀雅洋, 溝口理一郎, 河内雅亮, 上原邦昭, 豊田順一, 角所収:

電子情報通信学会論文誌 (D), J71-D, 5, pp.774-781 (1988)

1. **A Voice Input-Output System Using Island Words**
Abudui Mobin, S.S.Agrewal, Anil Kumar and K.D.Pavate:
Proc.ICSLP'90, Paper 7.22, Kobe, vol.2, pp.1381-1384, 1990
2. **Evaluation of a Speech Understanding System -SUSKIT-2**
Yutaka Kobayashi and Yasuhisa Niimi:
Proc.ICSLP'90, Paper 17.8, Kobe, vol.2, pp.725-728, 1990
3. **Spoken Language System Integratin and Development**
Patti Price, Victor Abrash, Doug Appelt, John Bear, Jared Bernstein, Bridget Bly,
John Butzberger, Michael Cohen, Eric Jackson, Robert Moore, Doug Moran, Hy Murveit
and Mitchel Weintraub:
Proc.ICSLP'90, Paper 17.9, Kobe, vol.2, pp.729-732, 1990
4. **Integration of Speech Recognition, Text-to-Speech Synthesis, and Talker
Verification into a Hands-Free Audio/Image Teleconferencing System(HuMaNet)**
D.A.Berkley and J.L.Flanagan:
Proc.ICSLP'90, Paper 20.1, Kobe, vol.2, pp.861-864, 1990
5. **Media Conversion into Language and Voice for Intelligent Communication**
Fumio Maehara,Masamichi Nakagawa, Kunio Nobori, Toshiyuki Maeda,
Tsutomu Mori and Makoto Fujimoto:
Proc.ICSLP'90, Paper 23.5, Kobe, vol.2, pp.985-988, 1990
6. **A Knowledge-Based Understanding System for the Chinese Spoken Language**
Yanghai Lu and Beiqian Dai:
Proc.ICSLP'90, Paper 27.9, Kobe, vol.2, pp.1221-1224, 1990
7. **Conversational Speech Understanding Based on Cooperative Problem Solving**
Akio Komatsu, Eiji Oohira and Akira Ichikawa:
Proc.ICSLP'90, Paper 27.10, Kobe, vol.2, pp.1225-1228, 1990
8. **Heuristic Search Problems in a Natural Language Task Oriented Spoken
Man-Machine Dialogue System**
P.Mousel, J.M.Pierrel and A.Roussanaly:
Proc.ICSLP'90, Paper 27.13, Kobe, vol.2, pp.1237-1240, 1990
9. **Tools for Designing Dialogues in Speech Understanding Interfaces**
Anders Baekgaard and Paul Dalsgaard:
Proc.ICSLP'90, Paper 27.16, Kobe, vol.2, pp.1249-1252, 1990

10. A Speech Understanding System

Spriet Thierry:

Proc.ICSLP'90, Paper 27.20, Kobe, vol.2, pp.1265-1268, 1990

11. SPICOS II-A Speech Understanding Dialogue System

Harald Höge:

Proc.ICSLP'90, Paper 29.5, Kobe, pp.1313-1316, 1990

12. Presuppositions and Anaphora in a Question Answering Speech System

M Streit:

Proc.Eurospeech'89, Paris, vol.1, pp.175-178, 1989

13. The Integration of Speech and Natural Language in Interactive Spoken Language Systems

P Price, R Moore, H Murveit, F Pereira, J Bernstein and M Dalrymple:

Proc.Eurospeech'89, Paris, vol.1, pp.179-182, 1989

14. Co-operation and Representation of Syntactic-semantic and Pragmatic Knowledge in a Natural Language Task Oriented Spoken Dialogue System

P Mousel, J M Pierrel and A Roussanaly:

Proc.Eurospeech'89, Paris, vol.1, pp.183-186, 1989

15. A Linguistic Knowledge Base for Applying Semantic Information to a Speech Understanding System

N Inoue, T Morimoto and K Ogawa:

Proc.Eurospeech'89, Paris, vol.1, pp.194-197, 1989

16. A Dynamic Dialog Model for Human-Machine Communication

D Luzzati:

Proc.Eurospeech'89, Paris, vol.2, pp.577-580, 1989

17. Conversations with Computers: Problems of Feedback and Error Correction

C R Frankish:

Proc.Eurospeech'89, Paris, vol.2, pp.589-592, 1989

18. Simulation Experiments and Prototyping of User Interfaces in a Multimedial Environment of an Information System

L Hitzenberger and H Kritzenberger:

Proc.Eurospeech'89, Paris, vol.2, pp.597-600, 1989

19. Syntactic Analysis in Speech Understanding

G Th Niedermair:

Proc.Eurospeech'87, Edinburgh, vol.1, pp.5-8, 1987

20. Using a Speech Synthesis System for Casual Conversation-a Serious Application

N Alm, A F Newell and J L Arnott:

Proc.Eurospeech'87, Edinburgh, vol.1, pp.353-356, 1987

21. Multi-Talk,a New Portable Multi-Lingual Speech Output Communication A

K Galyas and J Liljencrants:

Proc.Eurospeech'87, Edinburgh, vol.1, pp.357-360, 1987

22. Communication Aids and Voice Synthesis

B Brophy, J L Arnott and A F Newell:

Proc.Eurospeech'87, Edinburgh, vol.1, pp.361-364, 1987

23. Effective Human-Machine Interfaces for Use in Industry and Commerce

P Kelway:

Proc.Eurospeech'87, Edinburgh, vol.2, pp.206-209, 1987

24. AUDIOTEX:Computer Input and Output by Voice Telephony

R I Anderson:

Proc.Eurospeech'87, Edinburgh, vol.2, pp.253-255, 1987

25. Continuous Speech Understanding by Keyword Extraction in a Voice-mail System

Y Ariki, H Ohkawa and T Sakai:

Proc.Eurospeech'87, Edinburgh, vol.2, pp.256-259, 1987

26. A Knowledge-Based Approach to the Design of a Man-Machine Dialog System by Voice

N Carbonell, J-P Haton and J-M Pierrel:

Proc.Eurospeech'87, Edinburgh, vol.2, pp.357-360, 1987

27. Description of Task Dependent Knowledge for Speech Understanding System

T Kobayashi and K Shirai:

Proc.Eurospeech'87, Edinburgh, vol.2, pp.405, 1987

28. On the development of a phone communication aid for the hearing impaired

D.Kanevsky, P.Gopalakrishnan, C.Danis, G.Daggett, E.Epstein, D.Nahamoo:

Proc.Eurospeech'91, Paper 11.9, Genova, vol.1, pp.303-306, 1991

29. A communication system for the disabled with emotional synthetic speech produced by rule

I.R.Murray, J.L.Arnott, N.Alm, A.F.Newell:

Proc.Eurospeech'91, Paper 11.11, Genova, vol.1, pp.311-314, 1991

- 30. Tools and processes for developing low-cost and high-quality text-to-speech synthesis for communication aids**
R.W.King:
Proc.Eurospeech'91, Paper 12.3, Genova, vol.1, pp.325-328, 1991
- 31. Stochastic representation of semantic structure for speech understanding**
R.Pieraccini, E.Levin:
Proc.Eurospeech'91, Paper 13.1, Genova, vol.2, pp.383-386, 1991
- 32. An adaptable man-machine interface using connected-work recognition**
H.Bergmann, H.H.Hamer, A.Noll, A.Paeseler, H.Tomaschewski:
Proc.Eurospeech'91, Paper 17.2, Genova, vol.2, pp.467-470, 1991
- 33. A man-machine dialogue system for speech access to E-mail information using the telephone: implementation and first results**
P.Baggia, A/Ciaramella, D.Clementino, L.Fissore, E.Gerbino, E.Giachin, G.Micca, L.Nebbia, R.Pacifici, G.Pirani, C.Rullent:
Proc.Eurospeech'91, Paper 30.6, Genova, vol.2, pp.865-868, 1991
- 34. A.R.E.S.:an interface for automatic reporting by speech**
G.Antoniol, F.Brugnara, F.Dalla Palma, G.Lazzari, E.Moser:
Proc.Eurospeech'91, Paper 34.8, Genova, vol.2, pp.973-976, 1991
- 35. Operational validation of syntactic-semantic models in a spoken man-machine dialogue system**
G.Deville, P.Mousel:
Proc.Eurospeech'91, Paper 52.3, Genova, vol.3, pp.1477-1480, 1991
- 36. Building a user interface for speech recognition-based telephone application system**
A.Choinière, R.Descout, J.M.Robert:
Proc.Eurospeech'91, Paper 52.9, Genova, vol.3, pp.1503-1506, 1991
- 37. A Network Model Dealing with Focus of Conversation for Speech Understanding System**
T.Kobayashi and K.Shirai:
Proc.ICASSP86, Paper 30.8, Tokyo, vol.3, pp.1589-1592, 1986

自動翻訳電話の動向と課題

樽松 明

(株)エイ・ティ・アール自動翻訳電話研究所

1. まえがき

自動翻訳電話とは、電話による通信の内容を双方の言語に自動的に変換するシステムで、様々な言語を持つ外国の人とのコミュニケーションを言葉の違いを感じないでスムーズに行なうものである。自動翻訳電話システムの構築のためには、話し手の音声聞き分けて認識する音声認識、異なる言語間の変換を行う機械翻訳、計算機から音声を作り出す音声合成の要素技術が必要で、これらを総合したシステムとして実現される。

自動翻訳電話システムに対するニーズは、我が国の経済、文化の面での国際化が著しく広がるにつれて、大きくなっている。言語が異なる外国とのコミュニケーションの場面で、どのような言語を用いて意思疎通をするかが問題となってくる。日本語は、言語や構造が、欧米の主要外国語と大きく異なるため、言語の孤立性が著しく、日本人の中に外国語に対してハンデイキャップを感じる人が多い。一方、外国の人からみても、日本語は、非常にむずかしいということになっている。このため、将来の夢として、新しい情報処理技術を駆使して、世界の人々とのコミュニケーションを言葉の違いを感じないで行ないたいとの期待が大きい。

夢の自動翻訳電話システムが実現される可能性は、コンピュータ技術の進歩、および、音声認識、機械翻訳、音声合成の各要素技術の高度化の進展による。本稿では、自動翻訳電話について、関連する技術の研究動向をのべる。

2. 自動翻訳電話のイメージ

自動翻訳電話システムは、図1に示されるイメージのように構成される。

図1は、たとえば、日本語と英語との間の自動翻訳電話のイメージを示す。日本語の話し手が、「モシモシ」と話すと、日本語音声認識により音声文字の文章にかわる。日英機械翻訳により、日本語の文章は英語の文章に変換される。英語音声合成により、英語の音声で、「Hello」と出力される。逆に、英語の話し手が、「Yes, That's right.」と話すと、同様な処理によって、日本語の音声で、「ハイ、ソウデス」と出力される。

このように、自動翻訳電話は、言葉の違う人の間に、自動翻訳システムをおき、相互にコミュニケーションするものである。話し手と聞き手の二人の利用者が、対話における実際の参加者である。聞き手は、翻訳された音声聞いて、話し手の意図を理解することになる。この点は、人間と機械とのコミュニケーションとは異なる点である。人間である聞き手による理解力や常識の助けによって、機械翻訳システムの不十分な点が補われて、自動翻訳電話システムが利用されることが期待できる。

音声認識および機械翻訳の性能が十分に高くなければいけない。システムの入力と出力が音声になるので、機械翻訳においては、通常の文書を対象とする場合に行われる前処理や後処

理が期待できない。異なる言語の話し手と聞き手が、困難なくスムーズに会話をやりとりできるように、機械翻訳した結果は、聞き手にとって十分に理解できる品質でなければならない。

3. 自動翻訳電話技術の研究動向

自動翻訳電話システムの研究開発には、音声認識、機械翻訳、音声合成の要素技術の高度化をはじめ、種々の音声処理と言語処理の統合化など課題を解決しなければならない。現在、これらの技術については、それぞれあるレベルまでの研究開発が進んでいるが、自動翻訳電話システムの実現には、まだまだ不十分なレベルで、基礎的な研究の積み重ねが必要である。

自動翻訳電話は、1983年のテレコム83において、将来の夢の技術のラボラトリモデルとして、日本電気(株)が非常に簡単な音声翻訳システムを紹介し、注目を集めた。その後、自動翻訳電話のための基礎研究を行うため、1986年にATR自動翻訳電話研究所が設立された。以来、我が国では、世界に本格的な研究が開始され、音声認識、機械翻訳、音声合成などの要素技術の高度化の研究が進められている(1)。

英国British Telecom研究所では、自動翻訳電話システムの前段階の簡単なプロトタイプの例として、単語音声認識装置と音声合成装置を組み合わせ、英語とフランス語との間の音声翻訳実験をしている(2)。旅行予約のように限定されたタスクで、400種類の短文を選んでおき、それぞれの短文に定めた、3個程度のキーワードを音声認識して、短文を判定し、その短文番号を相手に送り、合成側では、相当する文を音声合成するものである。

端末キーボードを用いた翻訳文字電話システムとして、1987年に(株)東芝は、文書翻訳用の日英/英日機械翻訳装置を通信回線で接続し、簡単な文の実時間翻訳の実験を行った(3)。

カーネギーメロン大学(CMU)では、音声認識と自然言語高速処理方式を結合して、医者と患者の会話で小規模な音声翻訳システムの実験を行い(4)、1988年には、我が国の松下グループが、CMUの翻訳システムに音声認識装置と音声合成装置を接続して、50語程度の小規模な音声通訳装置の実時間システムのデモンストレーションをした。さらに、本年はじめには、日本電気(株)が、コンサートチケット予約における定形的な簡単な構文で、数百語の小規模なもので、日英/英日の双方向の音声自動通訳システムのデモンストレーションを行っている。これらは、小規模なものであれば、音声通訳システムの実現可能性が期待できるとして、関心を集めている。

そのほか、米国では、DARPA計画により、CMU、MIT、SRIなどが中心となって、会話音声認識の研究が進行中であり、飛行機の予約という限定した分野で、音声処理と言語処理の統合化による会話音声システムの研究開発が進んでいる。また、ATTベル研究所、IBMなどでは、高精度の会話音声認識の研究が継続されている。

ヨーロッパにおいても、音声認識、音声合成、機械翻訳のそれぞれの研究開発が進められている。EC諸国の共同により、英語、フランス語、ドイツ語、イタリア語などのそれぞれの言語で、それぞれ切符の予約やデータベース検索などの限定した分野で、音声認識による質問応答に関する研究開発が実施されている。

自動翻訳電話システムの究極の目標は、分野を限定せずに、自由な会話を自動翻訳することであろう。しかしながら、現時点でみて、実現可能な技術レベルでは、限定したタスクオリエンテッドな分野において、利用されることとなろう。会話の内容も、目標に向けたものを扱うというように、制限されたものとなる。自動翻訳電話の研究開発は、当面現実的な応用を念頭におきながら、限定した分野での適用を目指して進められている。以下、音声認識、機械翻訳、音声合成のそれぞれの要素技術の研究概要をのべる。

4. 音声認識

人間が発生する音声信号は、言語にもとずいて言葉が発声されるので、言語辞書項目にない言葉は出現しないこととか、舌や口の動き方の制約から極端に難しい発音はおこなわれないことなどの特性を持っている。会話音声は連続な発声であるので、連続音声をいかにして認識するかが課題となる。また、自動翻訳電話には、数千から数万の大語彙の音声認識技術が必要である。語彙の数が多くなると、同じ音素の波形でも発声者の話す速度や話す状況によりもともと変形が大きいのに加えて、多数の類似した音声(例えば、かいりょう(改良)とたいりょう(大漁)など)を識別しなければならなくなるので、複雑な処理が必要となる。

音声認識では、発声する音声の内容と発声のしかたの違いによって、認識できる語彙数が異なる。図2に、発声モードと認識語彙数について、現在の音声認識技術レベルを、特定話者と不特定話者にわけて示す。

4.1 大語彙連続音声認識

大語彙の連続音声認識の研究は、音韻を基本単位として認識し、これから単語や文節の認識へ積み上げていくアプローチで進められている。ATR自動翻訳電話研究所では、音韻を単位とした大語彙連続音声認識の研究を進めており、日本語連続音声認識技術において、目覚ましい進展がみられている。

音韻の認識には、統計的アプローチとして“Hidden Markov モデル(HMM)”が現在最も強力で、音韻モデルを精密、かつ頑健にした音韻認識方式が確立されてきている。HMM法によるモデルは、いくつかの状態と状態間の遷移を表す弧から構成され、各弧には状態間の遷移確率と入力コードに対する出力確率を持っている。Hidden Markovモデルでは、たとえば子音の/d/は、スペクトルの的に分析すると3つの状態があり、各ループには継続時間の統計により、この状態を確率的に移っていくとみなしている。日本語連続音声を認識するために、40種類の音韻HMMモデルを設定する。

連続音声の認識を容易に行うために、文節毎に区切って発声することとして、文節単位の連続音声を、言語的な予測を合わせて効率よく処理する方法がATRで開発されている。HMM音韻モデルと言語予測を行うLRパーザ(Left to Right パーザ)を結合して、連続音声を効率よく認識する方式(HMM-LR方式)を実現した(5)。まず、日本語の音韻毎にHidden Markovモデルを作成しておく。LRパーザとよばれる高速単語予測解析手法により、次にくる音韻を予測し、日本語の音韻毎に作成しておくHMMによる確率値を認識スコアとしてもどすことにより、連続音声の認識が可能となる。言語的な制約は文法規則で記述する。図3に、HMM-LRの構成を示す。

文法規則はLRテーブルに変換しておくため、高速な処理ができる。予測LRパーザでは、LR表の横一行の探索によって、音韻予測を行うことができる。たとえば、/mochiru/という音声の認識で、言語的な制約で、/mochiiru/と/motsu/と/morau/の3つの語が可能である例として、/mo/のあとで、/ch/と/ts/と/r/の音韻を予測し、それぞれの認識確率を出し、高い確率のものをとる。すなわち/mo/のあとに/ch/と/ts/を連結し、次の音韻の処理にうつる。このようにして文節のモデルを合成していく。

1000単語程度のタスクで、特定話者の場合で、88%の文節認識率を達成し(6)、上位5位までに正解が入っている認識率は99%であった。この方式により、ハードウェアの試作が進められている。

4.2 不特定話者音声認識

音声認識の厄介な点は、人によって音声の性質が違ふことである。一般の利用者が自由に

利用するために、男女、老若など多種多様な話者の音声に適用できるような不特定話者音声認識技術が必要である。大語彙でかつ不特定話者の音声認識技術は、現状ではまだ困難である。話者に対する対応をスムーズに行うには、話者に依存した情報を認識システムに取り込むという話者適応方式が有効である。この場合、認識しようとする全ての語彙を発声してシステムに登録することは、使用する人の負担が大きすぎて現実的でない。このため、事前に発声した少量の音声データ、およびシステム内の情報に基づいて、話者適応化をするアルゴリズムが研究されている。

標準話者と発声者のスペクトルを各々、256個程度のスペクトルで近似する。その上で、図4に示すような発声者の符号帳のスペクトルと、標準話者の符号帳のスペクトル間の対応関係を見い出す。

話者適応アルゴリズムを高度化し、「いよいよ」「いきおい」などの25単語の発声で話者適応が行えるアルゴリズムがATR自動翻訳電話研究所で確立されつつある。この方式で、1000単語のタスクで79%の文節認識率が達成されている(7)。

4.3 会話音声認識

会話音声認識においては、音声認識と言語処理を統合して、音声認識に付随する曖昧さをいかに効率よく減少させ、最終的に正しい文を選択可能とするかが大きな問題となる。連続会話音声の場合は、単語や言葉の区切りが明確ではないので、構文や意味、さらに話の話題や、前後の文脈などの知識情報の助けを借りて、最適な認識をする必要がある。さらに、候補の数が組合せとして多数でてくるので、音声処理と言語処理のインターフェースをうまくとって計算処理量が少なくなるように効率的な解析方法が必要である。

これを解決する方法としては、言語特性のモデル(言語ソースモデル)化を行い、処理の各段階において最も有効な言語ソースモデルを利用して、出現する単語/音韻を予測することが重要である。この考えに基づき、部分的な言語モデルを段階的に適用することにより、高精度の音声認識を行い、また認識結果の候補のなかから最も正しいと思われる文を効率的に選択する方式が開発されている(8)。

5. 機械翻訳

機械翻訳の一般的構成を図5に示す。文の形には数多くの変化形があり、これをコンピュータで能率よく翻訳するために、解析、変換、生成という3つのプロセスからなる。

解析部では、まず、入力された文を単語単位に分割し、単語の品詞と活用形を調べる。つぎに、辞書と文法規則を適用して、単語同士の関係を解析し、全体として整合性がとれる文の構造と意味の表現形態を得る。変換部では、原言語の構造と意味の表現を、辞書と規則を用いて相手言語の基本的構造と意味の表現に変換する。言語による表現の違いは、この変換部によってうまく訳語が選択されるように処理される。生成部では、相手言語の構造と意味の表現から、相手言語の文をつくりだす。まず、文体などの構造をきめた後、相手言語らしい整形した文をつくる。

5.1 話し言葉の機械翻訳

自動翻訳電話システムにおける主な翻訳対象は、話し言葉会話文である。話し言葉会話文は、通常の文書とは用語的および文法的に異なっており、書き言葉には見られない種々の特有な言語現象がある。実際に収集した日本語の会話文の例の分析によると、次のような特徴があ

ることがわかった。

- (1)主語や、述語の一部などの省略が多い。(例:その周辺でお願いします。)
- (2)敬語用法が頻出する。(例:いらっしゃいますか?)
- (3)語順が多様である。(例:電話を太郎が妹にかけた。)
- (4)あいまい性が頻出する。(例:それ位ならいいです。)

さらに、会話では、お互いが共通な常識があるものという前提のもとに、言葉がやりとりされる。会話文は特に文脈や常識に依存しているものが多い。このため、これまでの文書を対象とした従来の解析方式では対応が困難である。従って、自動翻訳電話のための機械翻訳技術は、従来より質的に高度な言語処理が必要となる。大規模な会話文の解析の研究課題は、文を構成する要素(語彙)から語用論も考慮して正しく全体の意味を組み立てること、文脈の解析や常識などの知識ベースを利用して省略語などを推定すること、構文的に正しいとは言えない文も解析可能とすること、などであることが明らかになってきた。

話し言葉の翻訳には、話し言葉に対応できる解析手法が必要である。省略や語順の変化などにうまく対応できる翻訳をするために、言葉の意味をより深く処理した意味解析、文脈分析、および知識表現を利用した処理方式が必要である。

ATRでは、話し言葉に適した日本語解析法を提案し、高速な演算手法を実現した。書き言葉に比べて言語表現が複雑な話し言葉を翻訳するために、文法を簡単にし、辞書である語彙記述を豊富にした、素性構造にもとづく語彙主導型解析手法をとっている。辞書には、語と語が結合する関係を、素性とよばれる構成単位によって記述しておく。文を形態素解析によって単語に区分分けしたのち、単語の結合条件を調べ、単一化という操作によって単語の素性の結合をおこなう。

対話文の持ち得る様々なニュアンスも処理するという観点から、様相(ニュアンス)表現や敬語・受給・否定表現の言語解析に基づき、発話の意味を命題内容と意図に分けて取り扱う意図伝達翻訳方式を提案した⁽⁹⁾。図6は、対話文の日英翻訳実験システム(NADINE)の構成を示す。「会議に申込みたいのですが」という文を解析し、ニュアンスをあらわす意図を解析段階で抽出し、中間言語的表現で相手言語にうまく表現できるようにしている。様相(ニュアンス)表現については、文末の表現「たいのですが」をもとに、「願望」の意図を抽出している。発話意図のタイプには、「要求」、「報告」、「依頼」などがあり、意味的な分類ができることを、話し言葉の対話文において明らかにした。この枠組みをさらに展開して、日本語対話における待遇表現などの制約を用いて、省略されている主語を補完する処理を実現している。

端末間対話文を中心とした発話について文末表現のバリエーションを含む発話例を用いて解析実験を行った。初級レベルの文法の基本部分が作成できた、世界ではじめて本格的な対話翻訳が実現されている。

図7に、ATR自動翻訳電話研究所の対話文翻訳実験システムにおける翻訳結果の例を示す。

5.2 音声翻訳

日本語音声認識、英語への翻訳、音声合成までを一貫して行う日英音声言語翻訳実験システムを構築し、その動作実験がATR自動翻訳電話研究所で行われた⁽¹⁰⁾。本システムは、文節連続音声認識、文節候補間に係り受けチェック/制御、対話文日英翻訳、および英語音声合成装置(DECtalk)から構成される。図8は、SL-TRANSシステムの構成を示す。約400の語彙からなる国際会議問合せタスクに関する会話を対象としている。システム全体としての文認識率は92%であった。この実験の規模はまだ小さいが、音声認識から音声合成までを1つのシステムとして動作させ、かつ本格的な対話文を音声翻訳可能としたシステムとしては世界で初めてである。

6 音声合成

自動翻訳電話における音声出力には、任意の文の音声を合成できる技術が必要である。

6.1 自然性の優れた規則による音音声合成

音声合成の従来の手法では、音節などの固定したユニットをつなぎ合わせて音声を合成するため、接続の歪みや音節間の調音結合の不自然さがあり、明瞭かつ自然性のある規則合成音を得ることはむづかしい。音声の音響的、音声知覚的な面から望ましい音声単位を作成する必要がある、ATRでは大規模なデータベースから様々な音声単位を接続の容易性と調音結合の自然性を考慮して抽出し、この複合音声単位を用いて規則による音声合成の方式を新たに考案した(11)。図9に、複合音声単位を用いた音声合成の音声単位の例を示す。この方式が従来手法よりも明瞭性、自然性で優れていることが確認できた。

6.2 個人性のある声質

自動翻訳電話における合成音声では、話者らしさのある声で出力する、声質を変換する技術が重要である。声質変換には、話者適応のアルゴリズムとして開発した手法が適応可能である(12)。さらに、英語の規則合成器の音声を日本人の発声者の声質に変換する言語間にわたる声質変換実験も行い、その可能性が示されている。

7. 自動翻訳電話技術の将来展望

(1) 高度な音声認識

現在のところ、音声認識システムは、これを使う人が丁寧に話す必要があり、使い勝手の面で制限がある。音声認識の研究開発が進み、実用化されるようになり、人間が自由に話せるためのシステムの頑健さがますます重要になってきている。コンピュータ技術の進展により、複雑な計算をとまなう音声認識の処理も実時間で実現可能となる。個人差や発声条件などに依存する音声のゆらぎを統計的にモデル化することによって、音素認識の高精度が進み、21世紀初頭には、不特定話者で数千語の音声認識が可能となるであろう。今後は、マイクロエレクトロニクス技術の著しい進歩により、音声認識用プロセッサのボードがパソコンやワークステーションにとりつけられて、広い範囲で音声認識が利用されるようになるだろう。

(2) 高度な機械翻訳

機械翻訳で扱っていない言語現象はまだ非常に多い。図10に、機械翻訳の発展レベルを、翻訳のレベルと適用分野の大きさに関して、書き言葉翻訳と話し言葉翻訳にわけて示す。翻訳のレベルは、文法的に正しいもののみのレベルが最も基本的である。レベルが高くなるにつれて、論理的推論、意図、文脈、談話と高度な処理がなされる。代名詞が何をさすのか、語の意味をいかに深く取り扱うかなどは、文脈や談話知識の処理によって可能となる。

現在では、翻訳を一文単位でしか行っておらず、文と文との間の関係を考えていない。しかし、会話文では、前に話した内容やはっきりわかっている事は省略するので、文脈を考慮に入れた翻訳をする必要がある。断片的な言い方で文法的に正しいとは言えない文もある程度処理可能とすることなどが必要である。

機械翻訳で常に問題となるのは、どのような文を対象にするかである。文が含んでいる内容

には次の三つの要素がある。その一つは事実関係が主体なもの、もう一つは文を発話した話し手の人の気持ちや態度を含むもの、三つ目は発話された場や社会的な言語使用の要因を含むものである。現在では、第一の要素が主で、第二は一部だけである。第三の要素は社会常識などの知識ベースを用いる必要があり今後の課題である。

翻訳の対象をうまく限定し、その範囲での単語の意味や用法を精密に記述した文法規則と辞書を作成することにより、翻訳の質が高まるであろう。

(3) コンピュータ技術の進歩

機械による翻訳の結果には、文の形はなしているが、文法的に間違っているものやある部分が誤っているものなどがある。機械翻訳の質の向上のためには、翻訳における文法規則や辞書の精密化のための処理を施さなければならない。機械翻訳の性能は、コンピュータの演算速度とメモリ容量の進歩とともに上昇する。

今後のコンピュータ技術の進展予測によると、ワークステーションの処理速度は現在にくらべて、5年後に10倍、10年後には100倍それぞれ高速になり、メモリ容量は同様に、5年後に16倍、10年後には128倍になるとみなされている。また、パーソナルコンピュータは、5年後には現在のワークステーション以上の性能になると予想される。コンピュータの処理能力を、単位時間あたりの計算回数(MIPS)であらわすと、現在の大型コンピュータは平均20MIPS程度であるが、10年後には、1チップコンピュータで、この100倍の2000MIPSの能力を発揮すると予測されている。しかも、今後は、分散された多数のコンピュータが協調的に同時並列処理をおこなうことになるのでさらに性能アップが可能となる。

機械翻訳は、大規模になると文法や辞書が複雑になり、非常に巨大なソフトウェアシステムとみなされる。このようにコンピュータ技術の進展を見込むと、ある程度複雑な処理をともなう機械翻訳方式でも、言語理論に基づいてコンピュータアルゴリズムがきちんとしているものであれば、ハードウェアの性能向上によって容易に実現されよう。

自動翻訳電話の場合、実時間に近い高速処理と、聞き手が誤解をしない翻訳の質の高さが必要である。このための複雑な演算処理により、処理速度が犠牲になるが、翻訳の質と処理速度のバランスによって、機械翻訳システムの性能が発揮される。限定した範囲の会話文を対象にして、パーソナルコンピュータにて経済的に実現される可能性は大きい。

(4) パーソナル自動翻訳電話システム

自動翻訳電話システムは、パーソナルコンピュータの発展と協調分散システムの展開にあわせて、異言語間の知的言語処理システムとみることができる。音声認識については、個人の音声特徴をあらかじめ学習により適応化することにより、精度の高い音声認識が可能となる。機械翻訳については、翻訳の対象と期待する翻訳レベルがユーザによって異なるので、個人の用途分野にあわせてパーソナル自動翻訳電話システムが普及すると思われる。そのためには、個人の目的に適合するような、電子化辞書と自動翻訳ソフトウェアの充実が必要である。そのためには、さらに高度な機械翻訳に向けて、さらに基礎的研究を続けつつ、その成果を実際の機械翻訳システムに反映させていく必要がある。

8. むすび

音声認識および機械翻訳のそれぞれの技術が、いずれも人間が用いる会話音声および自然言語を取り扱うことになるので、種々の機能をそなえた、人間が使いやすいマンマシン機能を備

えたものでなければならない。これは、使用する人間と自動翻訳電話システムが互いに相補い、かなり人間がシステムに協力して利用していくというマンマシン共同システムが想定される。

自動翻訳電話システムは、通信ネットワークを通して、異言語間の通信を行う知的通信システムとみることができる。将来予想される通信形態は、そのメディアの多様性から、メッセージ翻訳通信、端末間対話文翻訳通信、通訳電話などが考えられる。これらは、技術レベルの進歩とともに、段階的に種々の展開がなされよう。

自動翻訳電話システムが実現する時期には、通信ネットワークはISDNによるデジタル化が進展し、音声やデータや画像を統合して経済的に通信できるようになろう。電話機の形態も、コンピュータを内蔵して、音声のみならずデータや画像もあわせて扱えるインテリジェント端末の形が多くなっているであろう。自動翻訳電話システムを利用する端末としては、音声、データ、画像を扱うマルチモードのインテリジェント端末を想定できよう。システムからユーザへの問い合わせ情報や、固有名詞など音声だけでは伝えにくい情報を、ディスプレイとキーボードを用いて、マルチモードでユーザとシステムとのインターフェースをとることが好ましい。このように、画像情報を相手に伝えて、画像を見ながら細かい会話をすると、お互いのコミュニケーションが容易になろう。

自動翻訳電話におけるそれぞれの言語に関する研究は、その言語を母国語とする国の研究が不可欠であるので、国際的な研究協力が望まれる。

[参考文献]

- (1) 樽松、“自動翻訳電話の研究動向”，信学誌、71, 8, pp.803-805 (Aug., 1988)
- (2) F.W.M.Stentiford and M.G.Steer: “A Speech Driven Language Translation System”, Proc. of European Conf. on Speech Technology, 1987, (Sept. 1987).
- (3) S. Miike, K. asebe, H. Somers, S. Amano, “Experiences with an On-line Translating Dialogue System”, Proc. of 27th Meeting of ACL, (June, 1988)
- (4) M. Tomita, M. Kee, H. Saito, T. Mitamura, H. Tomabechi: “The Universal Parser Compiler and Its Application to a Speech Translation System”, Proc. of Second Int. Conf. on Theoretical and Methodological Issues in Machine Translation”, (June, 1988)
- (5) 北、川端、斉藤: “HMM音韻認識と拡張LR構文解析法を用いた連続音声認識”, 情処学論, 31, 3, pp. 472-480, (March, 1990)
- (6) 花沢、北、中村、川端、鹿野: “HMM-LR音声認識システムの性能評価”, 音響誌、46, 10, pp.817-823, (Oct. 1990)
- (7) 中村、花沢、鹿野: “ベクトル量子化話者適応のHMM音韻認識への適応”, 音響誌、45, 12, pp.942-949, (Dec. 1989)
- (8) 柿ヶ原、森元: “SL-TRANSにおける文節候補の削減”, 第39回情処学全大, (Oct. 1989)
- (9) K. Kogure, H. Iida, T. Hasegawa, K Ogura: “NADINE An Experimental Dialogue Translation System from Japanese to English”, Proc. InfoJapan'90, Vol.2, pp. 57-64, (Oct., 1989).

- (10)T. Morimoto, H. Iida, A. Kurematsu, K. Shikano, T. Aizawa, "Spoken Language Translation: Toward Realizing an Automatic Telephone Interpretation System", Proc. of InfoJapan '90, vol.1, pp.553-560, (Oct., 1989)
- (11)Y. Sagisaka: "Speech Synthesis by Rule Using an Optimal Selection of Non-uniform Synthesis Units", Proc. of ICASSP-88, (April, 1988)
- (12)M. Abe, S. Nakamura, K. Shikano, H. Kuwabara: "Voice Conversion through Vector Quantization", Proc. of ICASSP-88, (April, 1988)

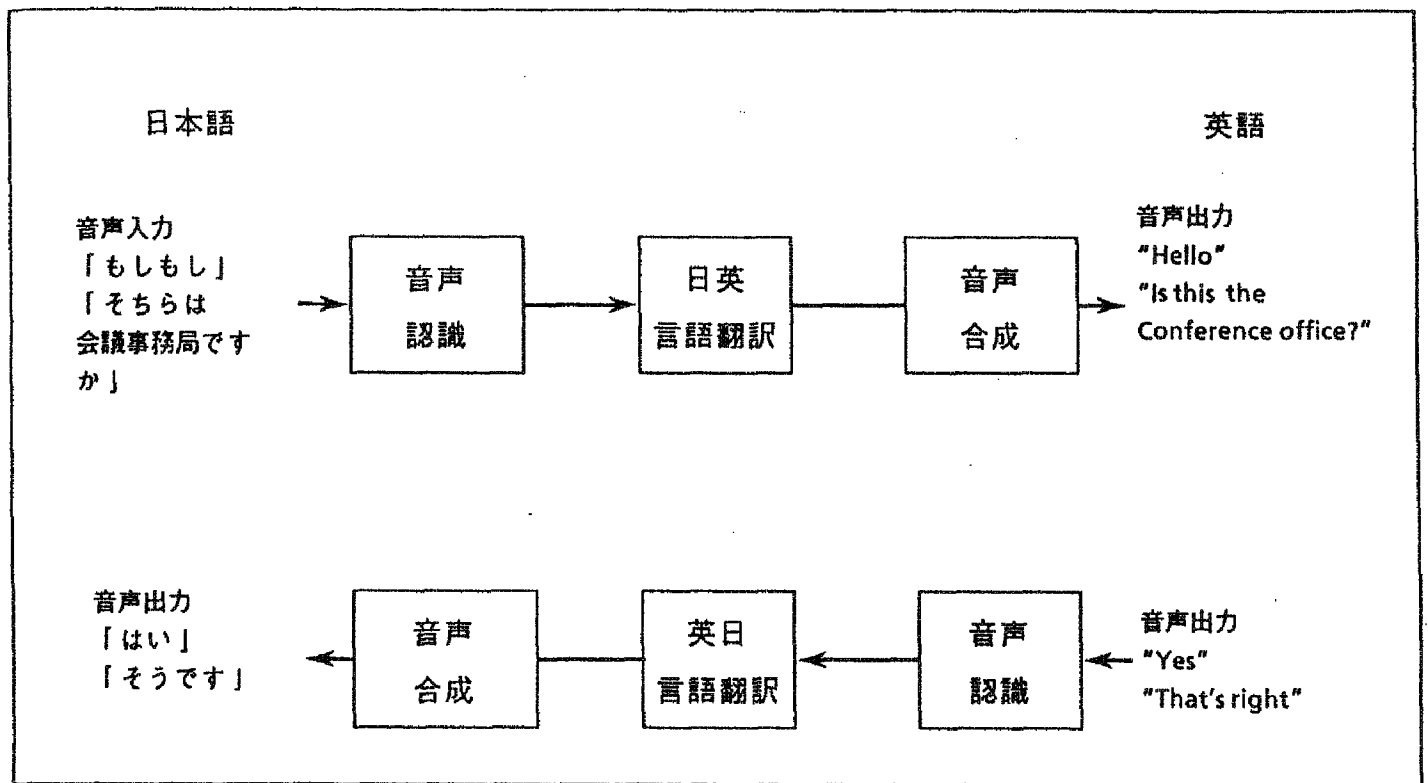


図1 自動翻訳電話のイメージ

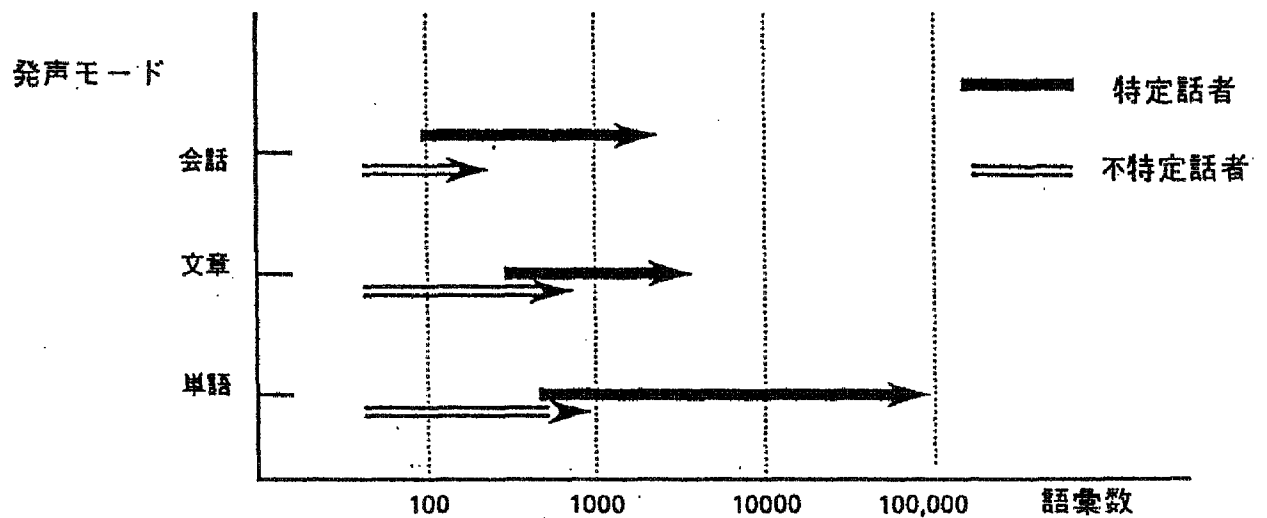


図2

音声認識における発声モードと認識語彙数

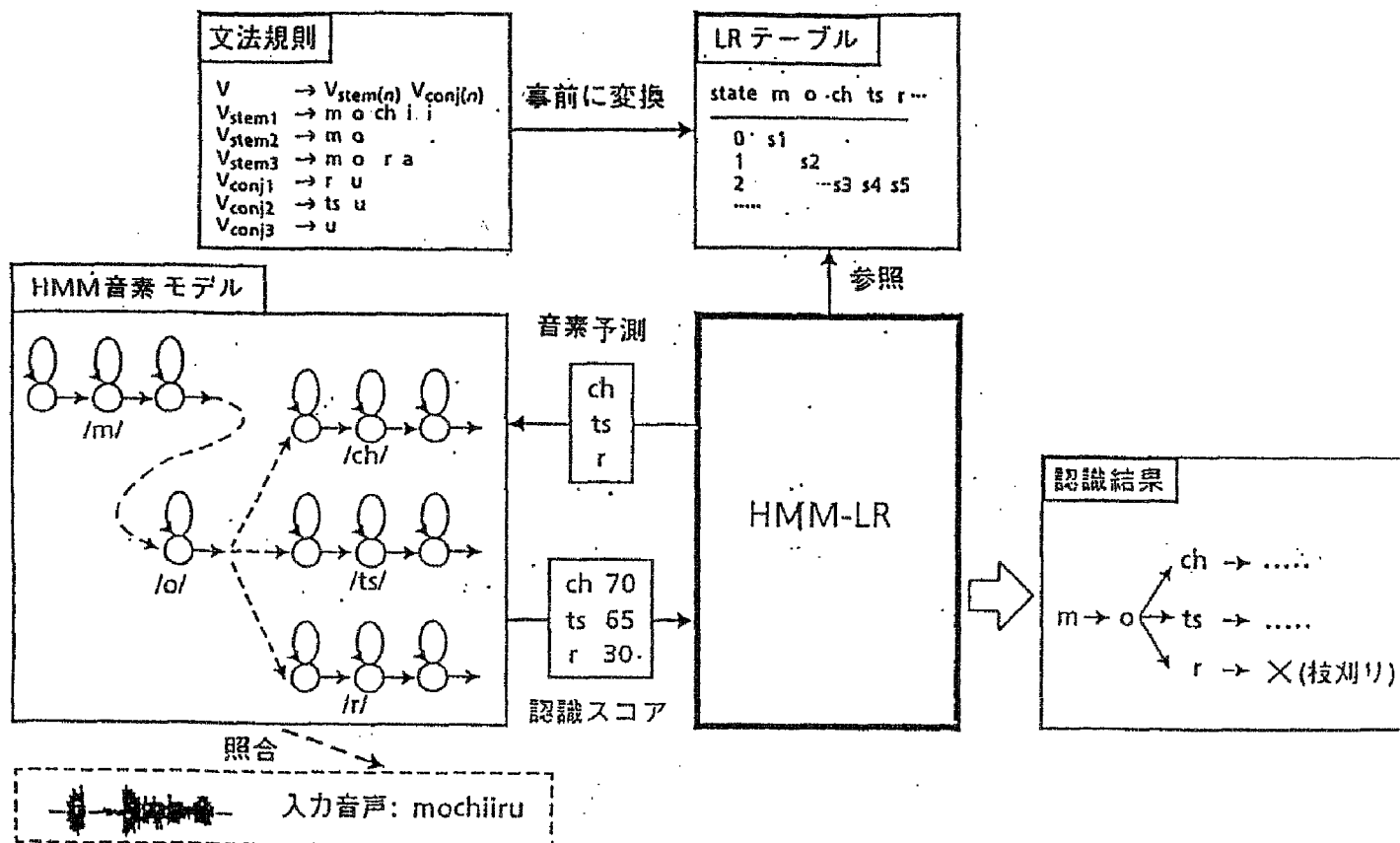


図3 HMM-LRによる音声認識

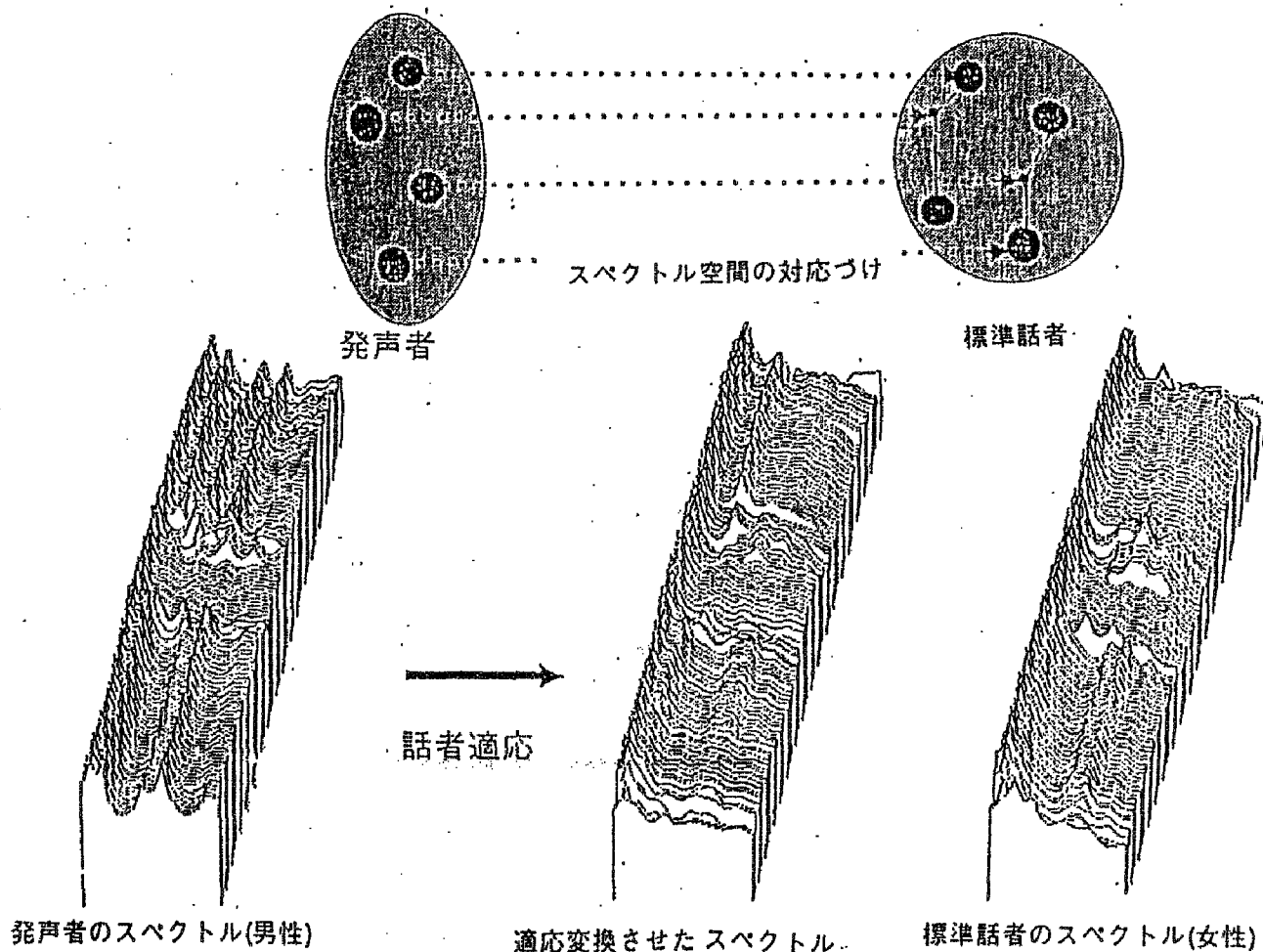


図4 話者適応の方法

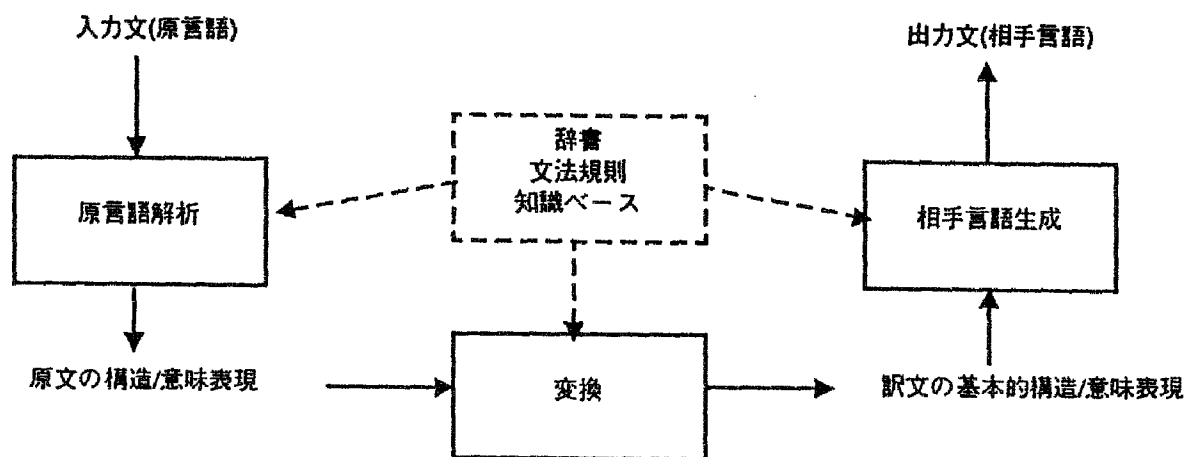


図5 機械翻訳の構成

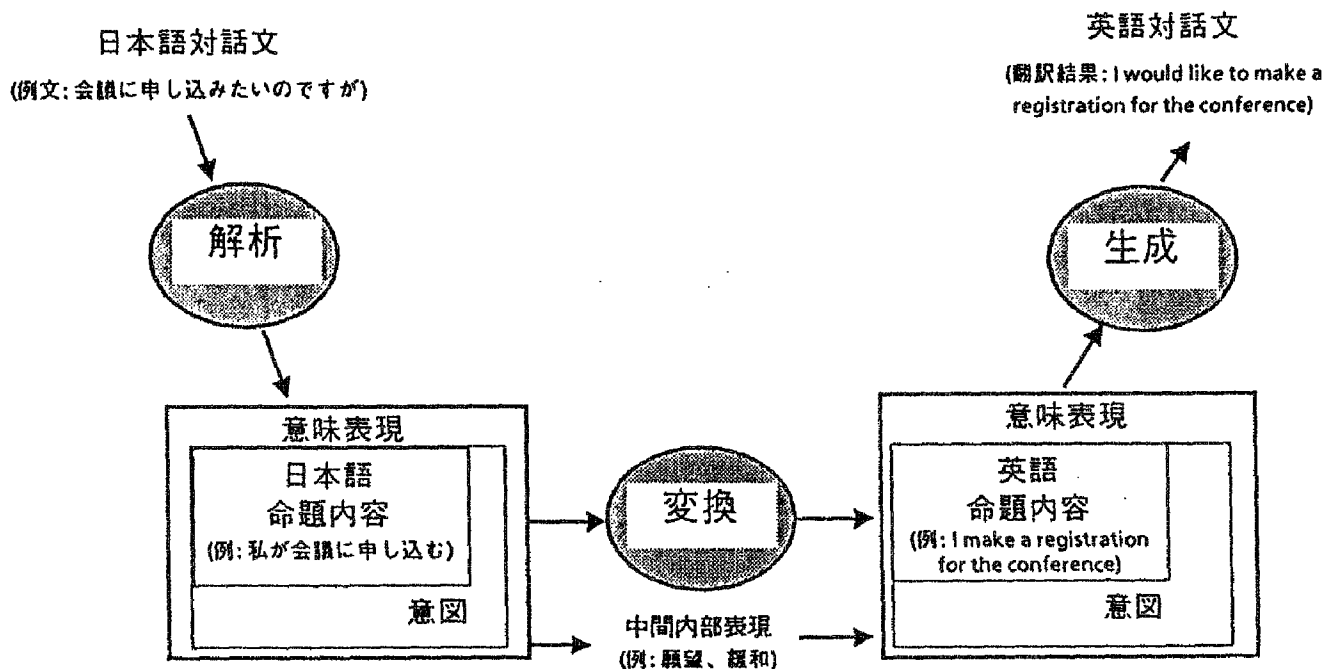


図6 対話文の日英機械翻訳の構成

もしもし
そちらは会議事務局ですか

Hello.
Is that the office for the conference?

はい そうです

Yes. That is right.

会議に参加したいのですが

I would like to take part in the conference.

まず、登録用紙で手続きをしていただかなくては
なりません

First, you must submit a registration form.

ご住所とお名前をお願いします

Could you give me an address
and a name?

わかりました

I see.

図7 対話文日英翻訳結果の例

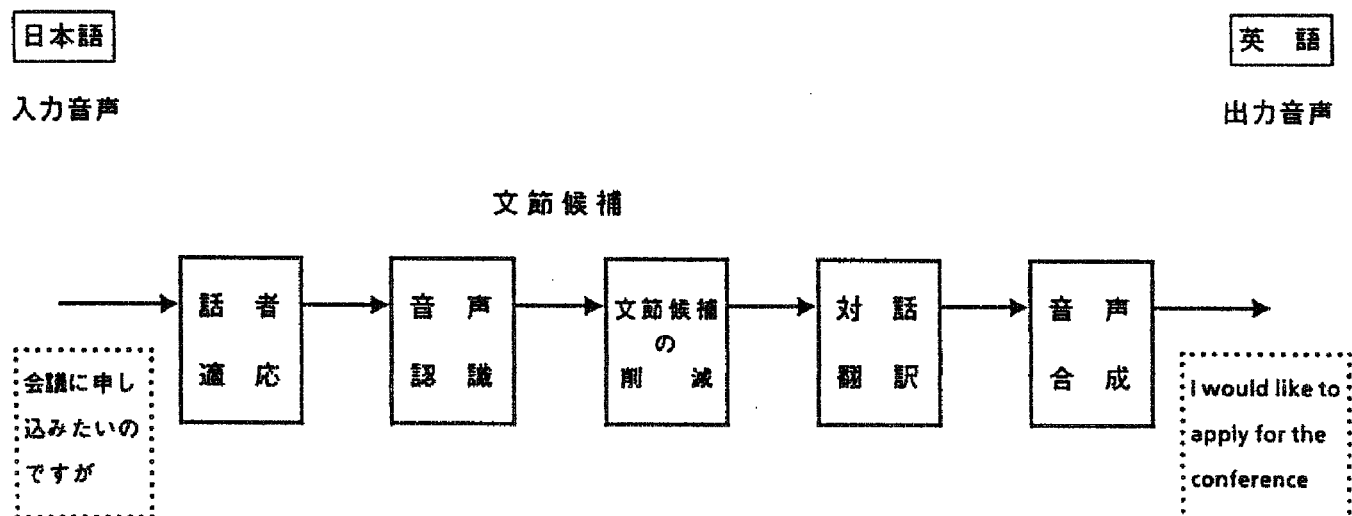


図8 日英音声言語翻訳実験システム (SL-TRANS)

/sochirawakaigijimukyokudesuka/
 (そちらは会議事務局ですか):入力音韻列

- ◎ 音韻環境に応じた接続
- ◎ 選択された単位に応じた接続

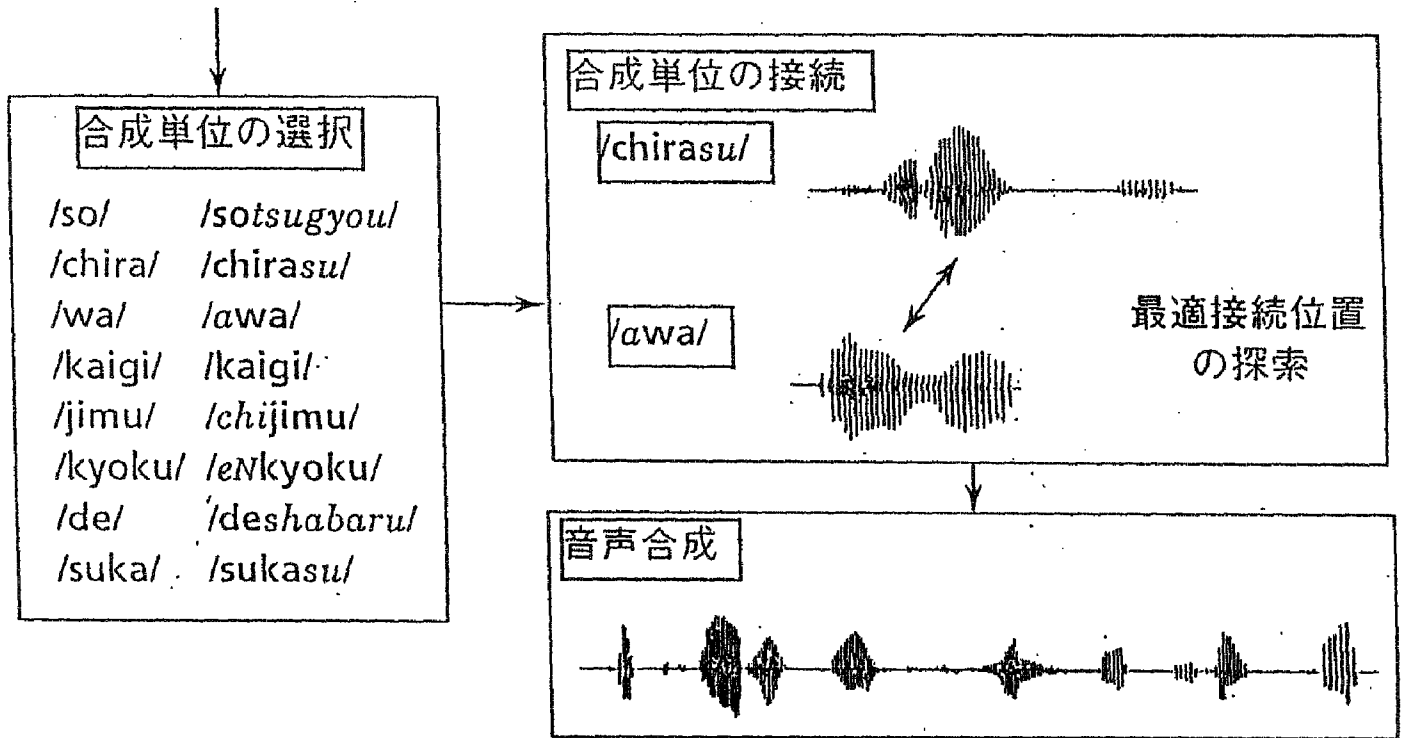


図9 複合音声単位による音声合成

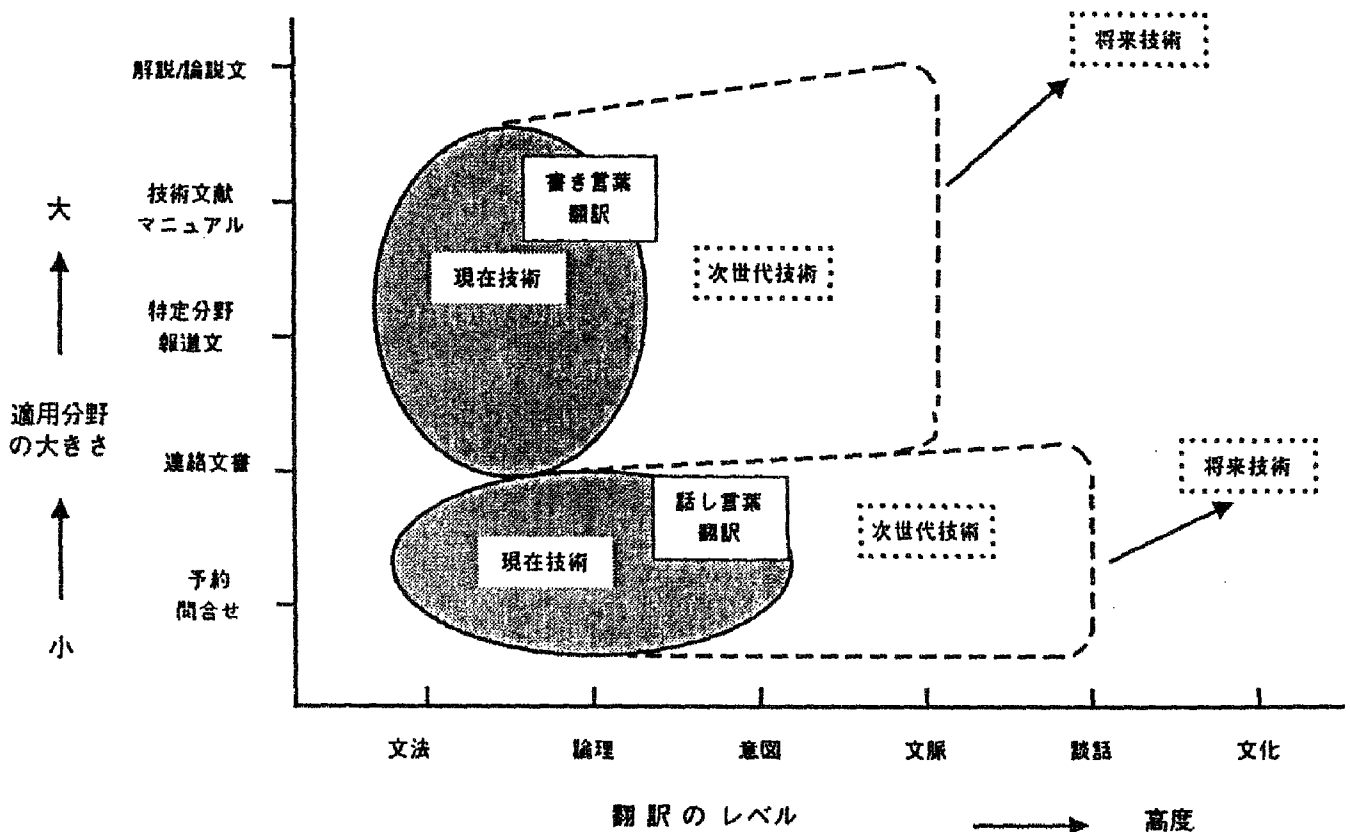


図10 機械翻訳の発展レベル

制約の力学

橋田 浩

(ICOT 第3研究室)

1 概論

認知過程における情報の流れは文脈に依存し、極めて多様である。したがってAIシステムにおいては、制約 (constraint)、つまり情報の流れを固定する度合の小さい設計方式を用いるべきである。学習は、システムの長期的属性の適応的な変化だから、長期的に保持される制約の変換である。また、刻々と変化する文脈に柔軟に対処するには、システムの一時的内部状態も情報の流れをなるべく固定しない形のものでなければならない。したがって推論もまた、次々に新たな制約を生成してゆく作業と考えられる。こうして、推論と学習をともに制約の変換としてとらえることができる。

言語使用などの組合せ的な行為を設計するには、1階論理程度の記述力を持つ制約の体系が必要だろう。ところで従来の記号論理では、制約が充足されている度合が充足と違反の2つに明確に区別されており、仮説が有意味であるためには充足されていることを保証せねばならない¹。しかしそれは、1階論理並みに強力な体系の下では実際上不可能である。認知主体が行なうべき情報処理は準最適解 (充足の程度が十分高い解) の効率的な探索と考えられるが、従来の記号論理ではそうした処理を設計できないのである。

そこで、連続値のポテンシャル・エネルギーとして違反の程度を計量することによってファジーな宣言的意味を与え、さらにこのポテンシャル・エネルギーの最小化によって情報処理を制御することを考える。このように単一の力学 (dynamics) に基づいて宣言的意味と操作的意味とを一挙に与えることにより、両者を密接に関連付けることができる。それに基づく情報処理は、意味的な重要性を反映し、仮説の生成や無矛盾性の管理のための膨大な計算を無闇に行なわないので効率がよく、また、制約違反によって停止/暴走することがないという意味で頑健 (robust) なものとなりうる。

力学によるアナログ情報処理の定式化は、ある種のニューラルネットなどで使われており、記号処理に関してもまた、動的計画法やゲーム木の探索など、評価関数の最小化という手法を用いることは多い。本稿は、これらのアプローチを一般的な枠組の下に統合する。つ

¹ 確率論理やファジー論理においては、真理値は確率的/ファジーであるが充足/違反はそうではない。非単調論理やデフォルト論理の場合は、充足/違反の概念に離散的な尺度を導入することもできようが、それでも許容可能な違反と完全な違反とは明確に区別される。

まり、アナログ的/記号的な推論/学習を全て制約変換としてとらえ、処理の制御を単一の力学から導出する方法を論ずる。

2 制約と力学

制約の記号的構造を節形式の1階論理プログラムとする。各節はリテラルの集合であり、各リテラルは要素制約 (atomic constraints) の前に正負の符号を付けたものとして表記する。要素制約には、 $p(X,Y)$ のような要素式と $X=Y$ のような等式がある。要素式は、 $X=f(Y)$ のような束縛を含む。等式には節の中の等式と外の等式とがある。節の中の等式は、たとえば節 $+p(X) -q(X)$ における X の2つの生起の間にある。つまりこの節は $+p(A) -A=B -q(B)$ と書いてもよい。節の外の等式は、単一化可能な2つの要素式の対応する引数同士を結ぶ。

各要素制約 α は、 $(0,1)$ 区間の実数である活性値 (activation value) x_α を持つ。 x_α は α が真である主観的確率と見なせる。要素制約の個数を n とすれば、制約全体のアナログ状態は、 n 次元ユークリッド空間の座標 \vec{x} である。制約全体のポテンシャル・エネルギー U は \vec{x} の関数である。論理プログラムの宣言的意味はいくつかの側面からなり、各々の側面が満たされていない度合が各種のエネルギーによって表現され、 U はそうしたエネルギーの総和である。

たとえば、 $-p +q$ という節における通常のリテラルの選言の意味は、下のエネルギーによって表現できる。

$$Dx_p \bar{x}_q$$

任意の値 v に対し、 \bar{v} は $1-v$ を表わす。 D はこの節に割り当てられた正の定数である。この式の値が小さいということは、 x_p が0に近いとか x_q が1に近い、つまり近似的には $\neg p$ か q が真だということである。また、同じ節におけるリテラルの相互排除の意味は、次のエネルギーによって表現される。

$$E\bar{x}_p x_q$$

E は正定数である。この式は、近似的に1つの節の中で真であるリテラルが一時に高々1個であるべきことを意味する。これによって、たとえば観測事実 q から仮説 p を導くというようなアブダクション (abduction) が起動される。

制約の宣言的意味には他に以下の4つの側面がある。要素式 α の完備化 (completion) とは、 α が真であるためには α と単一化可能なある要素式 β が真でなければならない、ということである²。要素式の同化 (assimilation) とは、2つの単一化可能な要素式が対応する位置に同一の引数を持つならば両者の真理値は近い可能性が高い、ということである。等式の推移律 (transitivity) は、等式の巡回路 (cycle) において、全ての等式の活性値が1に近いとか、2つ以上の等式の活性値が0に近い、ということの意味する。束縛の整合性は、等式の経路で結ばれた2つの引数が単一化不能な束縛の第1引数になっている (たとえば $A=f(X)$ 、

²Prolog などにおける完備化では、 α を節の本体の要素式に、 β を節の頭部の要素式に限定している。

$A=B$ 、 $B=C$ 、 $C=g(Y)$ という要素制約がある) とき、これらの要素制約のいずれかは偽だ、という意味である。 U には、これらの側面を表現する項の他に、各活性値を $(0, 1)$ 区間に正規化するための項があるが、委細は省略する。

3 推論

アナログ推論は、ある種のニューラルネットにおける推論とほぼ同様の活性拡散(spread-ing activation)である。活性拡散は、制約の記号的構造を変えずに活性値の割当てを準最適化して U の極小状態を求める演算であり、 U に関する最急降下法として定義できる。まず、現在の制約ネットワーク中の要素制約の個数を n とすると、制約ネットワークのアナログ状態は、下記のような n 次元ユークリッド空間の点である。

$$\vec{x} = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix}$$

この空間において、 U は下のような力の場 (field of force) を生ずる。

$$\vec{F} = -\text{grad}U = \begin{pmatrix} -\frac{\partial U}{\partial x_1} \\ \vdots \\ -\frac{\partial U}{\partial x_n} \end{pmatrix}$$

\vec{x} はこの力の場に従って下のように変化する。

$$\frac{d\vec{x}}{dt} = \vec{F}$$

この推論は論理プログラムの通常の意味に近い意味を反映しており、それによって、マーカ伝達 [1] や重み付きアブダクション [4] に相当する処理が自然に発生する [2]。

記号推論としては包摂化 (subsumption) を考えるが、その前に少し準備をしておく。まず、要素式や項や節をその具現例の集合と見なし、そうした具現例を世界の対象に対応させる解釈関数 (を具現例の集合に拡張したもの) を I とする。通常、 I の内容は処理係にとっては部分的にしかわからない。要素式 (または項) α と β が単一化可能であるとは $I(\alpha) \cap I(\beta) = \emptyset$ か否かがわからないということである。また、 $I(\alpha) \supseteq I(\beta)$ であることを α が β を包摂 (subsume) すると言う。

要素式 α が要素式 β を包摂化するという演算は、 β を含む推論規則 (節) を α に適用するということである。この演算は、 α と β が節の中で異符号のリテラルとして参照されているならば融合 (resolution)、同符号のリテラルとして参照されているならば因子化 (factorin) に相当する。ただし、これが通常の融合および因子化と異なるのは、 β を含む規則を具現化してできる規則に含まれる β の具現化 β' が、必ずしも α と単一化するわけではなく、 α に包摂されるに過ぎない、という点である。つまり、上記のように各要素式をその具現例の集合と見たとき、 $I(\alpha) \supseteq I(\beta')$ ではあるが、 $I(\alpha) = I(\beta')$ とは限らない。

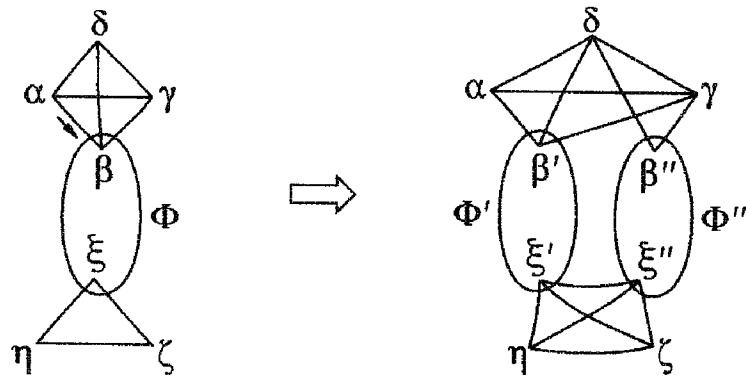


図 1: 要素式 α による要素式 β の包摂化

この包摂化の模様を図 1 に示す。ここで閉領域は節を表わし、ギリシア文字は要素式を表わし、結線は両端の要素式同士が単一化可能であることを示す。この演算は、 α に包摂されるような β の最大の部分集合 β' と $\beta'' = \beta - \beta'$ とに β を分割し、 α と β' の間に新たな包摂関係を生成するプログラム変換である。従って、 α と β' は β'' と単一化不能となる。また、 β の分割に伴って β を含む節が複写される。一方、 α と β' の間の同化を表わす U の項の係数に大きな値が設定されることにより、 α と β' の真理値を近付けようとする強い力が生ずる。

この他の記号演算としては、項に関する包摂化、および包摂化の逆の演算である消去 (deletion) もあるが、紙面の都合により省略する。

記号演算の優先度を、あるポテンシャル・エネルギー H の減少に対して期待される寄与の大きさとして定める。一般に H は、制約のトップレベルのエネルギーであり、生存の危機や不快を表わすと考えるのが適切だろう。また、記号演算によって制約の記号的構造が変化し、それにつれて U の定義も変わるが、 H の定義は変わらないとする。このとき、ある記号演算によって増大する U の係数を s とし、現在の \vec{x} の付近での H の極小値 P にこの演算がもたらす変化が大部分 s の変化によるものとすれば、この演算の優先度は $-\frac{\partial P}{\partial s}$ で評価できる。たとえば上記の包摂化の場合の s は、 α と β の間の同化のエネルギーの係数とすればよい。このように、記号演算の制御は P に関する最適化として定式化できるだろう。当然ながら、この方法がうまく行くためには、 $-\frac{\partial P}{\partial s}$ の大きな s を増大させる演算が実際に P を減少させることが多いように、エネルギー関数と記号演算とを設計することが肝要である。また、仮説の整合性の検査のためには、 $\frac{\partial P}{\partial s}$ の大きな s を増大させる演算を優先させることも必要であるから、記号演算の優先度は、結局 $|\frac{\partial P}{\partial s}|$ で計量されと考えられる。以上のようにして導かれる記号推論の制御法を近似するヒューリスティクスに基づいて、たとえば自然言語の文の処理を効率よく行なうことができる [3]。

エネルギー関数の係数の変化はニューラルネットにおいては学習であり、たとえば誤差逆伝播法を用いた学習は、階層型ネットワークにおいて、理想的な出力と実際の出力との 2 乗誤差としての H が小さくなるように U の係数を最適化する作業である。したがって、ここで記号推論の優先度を求める際にニューラルネットの学習のための手法が使える。実際、

相互結合型ニューラルネットワーク用に一般化された逆伝播[5]によって、上記の $\frac{\partial P}{\partial s}$ を効率よく計算することができる。

4 学習

学習もまた上記の P に関する最適化と考えられる。 H の定義を自由に変えられるとすれば、全く推論を行わずにいくらでも P を小さくすることができるだろう。したがって、 H の定義は「生得的」に決まっており、学習においては H の定義を(ほとんど)変えない範囲で制約が変換されると仮定する。

アナログ的な学習は、 U の中の係数を P に関して最適化することだと考えられる。つまり、係数 s を $-\frac{\partial P}{\partial s}$ に応じてゆっくりと変化させるわけである。こうしてたとえば、 P の減少に貢献する度合の大きい U の項の係数は大きくなり、 P の減少を妨げる度合の大きい項の係数は小さくなる。当然、この学習は制約の宣言的意味を変える。また、上述の記号推論の制御が正しく機能するならば、 P の減少に寄与する見込みの大きい記号演算の優先度がこの学習によって一層高められ、記号推論の高速化がなされる。因みに、通常の誤差逆伝播を用いた学習によって記号推論を高速化する研究[6]が既にあるが、そこで学習されるのは、宣言的意味論とは一応無関係な処理のヒューリスティクスである。

一方、記号推論は制約の記号的変換であるから、記号的な学習は、記号推論の結果の一部を何らかの形で長期的に記憶しておくことと考えられる。包摂化は新述語を生成することに注意せよ。たとえば、図1において、 β' と β'' とは単一化不能だから、それらの述語は異なると考えられる。 α がさらに γ と δ を包摂化し、 γ が α に包摂される最大の部分集合 γ' と $\gamma'' = \gamma - \gamma'$ に分割され、 δ も同様に δ' と δ'' に分割されたたすると、 α 、 β' 、 γ' 、 δ' の述語と β'' 、 γ'' 、 δ'' の述語とは別のものとなる。 P の減少に対する貢献の大きい節は消去せずに残しておくことにすれば、説明に基づく学習のようなことができると考えられる。2つの節を連結して1つの節にするような演算を考えれば、さらに学習の可能性は広がるだろう。記号的な学習に関しては、逆融合のような演算がいかんして実現されるかなど、他にも検討課題は多い。

参考文献

- [1] Charniak, E. (1986) 'A Neat Theory of Marker Passing,' *Proceedings of AAAI'86*.
- [2] Hasida, K. (1991) Dynamics of Symbol System - A Hybrid Architecture of Cognition - , 草稿.
- [3] Hasida, K. (1991) 'Common Heuristics for Parsing, Generation, and Whatever ...,' *Proceedings of the Workshop on Reversible Grammar in Natural Language Processing*.
- [4] Hobbs, J., Stickel, M., Appelt, D., and Martin, P. (1990) 'Interpretation as Abduction,' Technical Note 499, SRI International.
- [5] Pineda, F.J. (1988) 'Generalization of Backpropagation to Recurrent and Higher

Order Neural Networks,' in D. Z. Anderson (ed.) *Neural Information Processing Systems*, pp. 602-611.

- [6] Suttner, C. B. and Ertel, W. (1990) *Automatic Acquisition of Search Guiding Heuristics*, *Proceedings of the 10th International Conference on Automated Deduction (CADE)*, pp. 470-484.

音声入出力のための対話管理について

山下洋一 溝口理一郎

(大阪大学 産業科学研究所)

1. まえがき

今日、社会生活におけるコンピュータの利用が増大し、情報の収集や検索、種々の企業活動や教育などにおいて計算機を利用して問題解決を行なうシステム(問題解決器と呼ぶ)の研究が広く行われている。現在、これら問題解決器とユーザとの意思伝達手段としては、主にボタンやキーボードを用いて入力が行なわれ、ディスプレイ上に表示される文字テキストによって出力が行なわれている。しかし、ユーザの負担などを考えるとキーボードや文字よりも音声によって情報伝達が行なわれる方が好ましい場合も多く、マンマシンインタフェースとしての音声情報処理技術の確立が期待される。本報告では、このような機械との音声対話において対話管理の果たすべき役割とその実現に関して述べる。

2. 音声インタフェースにおける対話管理の役割

音声による機械との対話を人間同士の対話のように自然なものとしてインタフェース技術を確立するためには、多くの問題を解決しなければならない。この中で、対話に固有な処理として、語句の省略の同定・補充、利用者の意図の把握、意図を伝達するための最適な表現の決定などの問題を解決し、柔軟で高度な対話処理を行わなければならない。さらに、音声理解を助けるための次発話の予測、避けることができないと思われる音声認識誤りや合成音声の聞き誤りへの対応、出力音声の韻律的特徴の生成など、自然言語など他の媒体には見られない音声特有の問題も解決しなければならない。これらの問題は対話あるいは音声対話に固有なものであり、問題解決器に依存するものではない。従って、機械との音声対話を行なう時、このような問題は個々の問題解決器内部ではなく、それとは独立したモジュールで対処すべきであると考えられる。このような観点から我々は、図1に示すような汎用的な音声インタフェースの構成を目指している。

図1に示す音声インタフェースでは、まずユーザの発話が音響処理された後で音声理解システムに入力される。音声理解システムは対話管理部からのユーザ発話の予測を利用しながらユーザ発話を同定し、結果を概念表現で記述し問題解決器へ送る。問題解決器は、処理(推論)結果を概念表現で記述し対話管理部へ戻す。対話管理部は対話コンテキストに従って情報を概念表現に付け加え、それを概念からの音声合成システムSOCS[1]へ渡す。さらに、概念表現からの音声合成システムが音声を出力する。このような枠組における対話管理は、音声入力に関する処

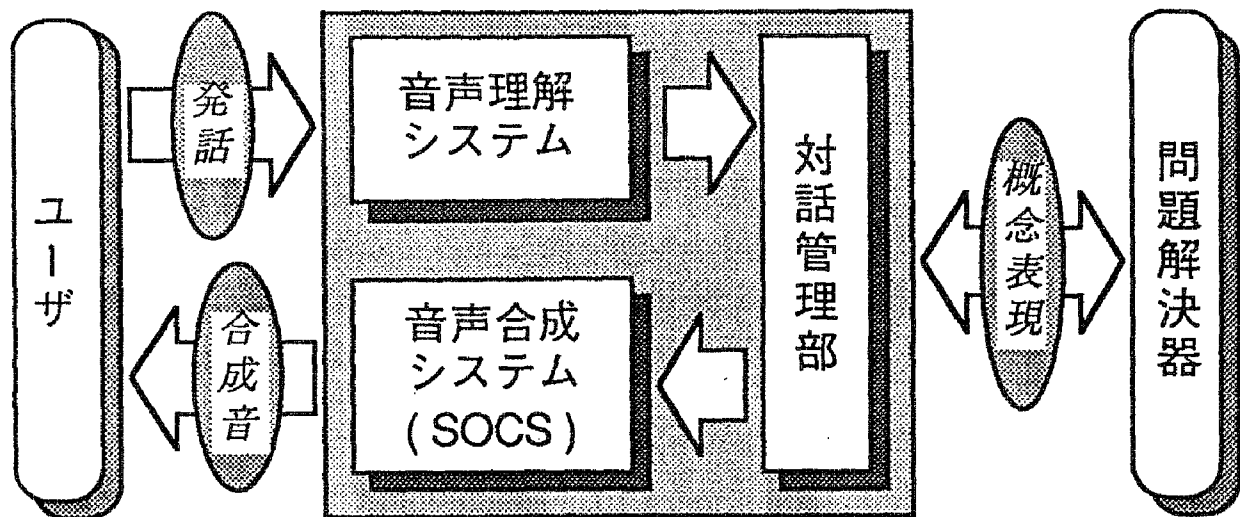


図1 音声インタフェース

理と音声出力に関する処理に大きく分けることができる。これを簡単にまとめてみると、

(1) ユーザから問題解決器への音声入力に関する処理

(1-1) ユーザ次発話の予測

音声理解システムを支援するために、対話コンテキストからユーザ発話を予測する。これには、対話が要求と応答のペアから構成されるという対話に関する知識や、問題解決器においてどのような話題が現われるかといったドメイン知識が利用できる。

(1-2) 問題解決器への情報の標準化

一般に、ユーザは同じ意味に対していくつかの異なる表現をとることができる。そこで、対話管理部がユーザの意図を判定し、一定の表現で問題解決器に情報を渡す。

(2) 問題解決器からユーザへの音声出力に関する処理

(2-1) 焦点情報の抽出

一般に人対人の対話では焦点のあたっている情報が省略されることが多いが、機械との音声対話では音声認識結果の提示と言う観点から、焦点情報を問題解決器からの発話に補うことも必要となる。また、焦点は概念表現からの音声合成にも重要な情報となる。

(2-2) 新情報の抽出

対話において新しく現れた情報を対話管理で管理し、音声合成システムで利用する。合成音の聴取では、始めて聞く言葉(単語)が聞き取りにくいことがあるので、例えば、発話速度をやや遅くするなどの処理が必要になる。

(2-3) 強調表現

問題解決器の発話において強調される語句の中には、対話のコンテキストから強調されるものが多い。対話管理部では、このような語句を抽出し音声合成システムに渡す。

(2-4) 慣用テンプレートへの変換

概念表現からの音声合成では、よく用いられる表現に対して韻律パターンを設定した慣用テンプレートを準備しておくことができる。問題解決器から入力された概念表現を慣用テンプレートを用いた表現に変換することにより、より高品質な合成音を出力することが可能となる。

となる。以下、これらの処理に関して順に述べていく。

3. 音声入力のための対話管理

3.1. SRプランに基づいたユーザ発話の予測

我々はこれまでに、MASCOTSと呼ばれる対話管理システムを構築し、SRプランに基づいた対話の理解とユーザ発話の予測について検討してきた[2]。そこでは、対話が主に要求と応答のやり取りから構成されることに着目して、このやり取りを17個のSRプランに分類した。17個のSRプランは要求を行なう対話者が問題解決器かユーザかで、それぞれシステムSRプランとユーザSRプランに分類され、実行時に活性化されたプランはシステム用とユーザ用の2本のスタックによって管理される。この枠組におけるユーザの次発話の予測は、プランの同定と密接に結び付いて行なわれる。ユーザが応答を返す場合には問題解決器の要求の発話によってすでに一つのシステムSRプランが活性化されており、このSRプランによってユーザの応答が予測される。ユーザが要求を出す場合には、音声理解システムに入力された文節ラティスの情報からボトムアップにどのユーザSRプランが活性化されたかを決定する。また、個々のSRプランにはユーザ発話を問題解決器に渡す時に取るべき表現が記述されており、問題解決器への情報の標準化がはかれる。

3.2. ドメイン知識を利用したユーザ発話の予測

従来のMASCOTSでは、要求と応答の組に基づいた対話の局所的な情報を用いてユーザの次発話を予測していた。しかし、ユーザの要求などのように局所的な情報だけでは十分に予測できない発話も存在する。そこで、Schankの提唱するMOP[3]の概念を導入し、対話の領域に依存する知識を用いたユーザ発話の予測を行なう[4]。

情報のやり取りに直接関与するユーザの発話は1)応答、2)情報の詳細化を行うための要求、3)不足情報を収集するための要求、4)評価、5)確認の計5種類に分類できる。一般に対話では、ユーザの次発話としてこれらのどの種類の発話も行うことが出来るため、次発話を予測する際には全ての発話の種類に関して発話文の構文や主題を予測しなければならない。ここで、返答の発話形式は直前の要求型の

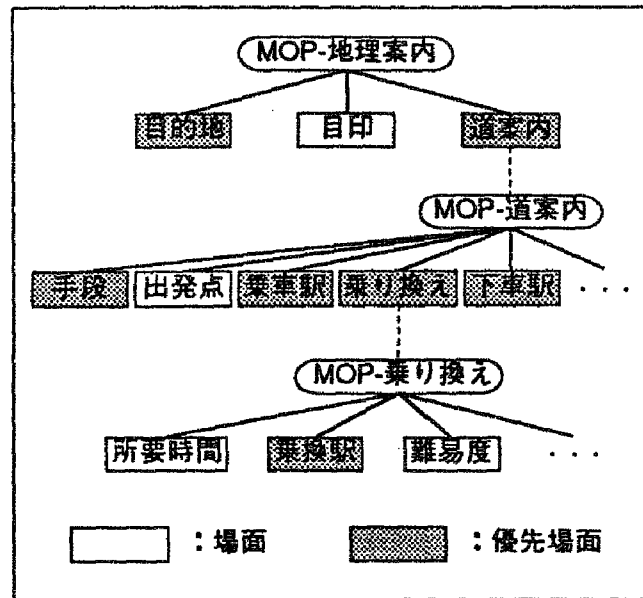


図2 MOPの例

発話の構文や意味内容に強く依存するという対話の局所的な性質から、1)、4)、5)のユーザの発話は、システムの発話からある程度予測することができる[2]。しかし、2)や3)の発話は、領域に依存する知識がなければ十分な予測を行うことができない。

3.2.1. MOP[3]

MOP (Memory Organization Packet) はある状況をいくつかの断片的な場面の集合として記述する概念であり、MOPを構成する各場面にはその場面において行われる事柄に関する知識が可能な限り一般的に記述されている。また、これらの場面は複数のMOPに共通の場面として利用することができるので対話の領域に依存する知識を効率よく記述することができ、さらに複雑な対話が行われる場面はより詳細な情報を記述した下位のMOPを持つ。ここで、図2に地理案内を行う際に用いられるMOPの例を挙げる。

図2のMOPは、「地理案内」という状況が、ユーザの目的地について対話を行う場面、案内するときに目印になるものについて対話を行う場面、そして実際に道を案内する場面の、三つの場面から構成されていることを示している。また図中の優先場面とは、対話に出現する確率が高く、その対話において重要な情報のやり取りが行なわれる場面を指す。

3.2.2. MOPを用いた発話予測

対話管理部はシステムの発話を受け取り場面の同定を行った後、上述したユーザの発話分類に基づいて発話予測を行う。まず、従来から用いているSRプランによって1)応答、4)評価、5)確認を予測する。次に、ドメイン知識に依存した予測として、2)情報の詳細化を行う要求、3)不足情報を収集するための要求の順に予測を行う。

S1: 東京まで新幹線で行って、地下鉄の丸ノ内線の新宿行きに乗ってください。

U2: 乗り換えは簡単ですか？

⋮

S9: 駅の出口を出て左へまっすぐです。

U10: 何か目印になるものはないですか？

(S: システム、U: ユーザ)

図3 対話例 (1)

情報の詳細化を行う要求は、ユーザが現在活性化している場面に関してより詳細な情報を要求する場合に行われる。従って、ユーザの次発話の主題は、その時活性化されている場面の下位の MOP を構成する場面に移動すると考えることができる。

一方、不足情報の収集を行う要求は、その時活性化している場面に関してユーザが十分な情報を受け取った後、別の場面に関する情報を要求する場合に行われる。従って、ユーザの次発話の主題は、その時活性化している場面と同じレベルに記述された他の場面か、あるいは上位の MOP に記述されている場面のいずれかであると考えることができる。但し、同じレベルにある場面の中では、常に優先場面が強く予測される。

3.2.3. 動作例

図2の MOP を用いて、ユーザと地理案内システムの対話 (図3に示す) を処理する際の発話予測について述べる。

まず、システム側の発話 s1 によって "MOP-道案内" に記述された "場面-乗り換え" が活性化される。この時予測される情報を詳細化するためのユーザの発話としては、活性化された場面の下位 MOP を構成する場面から、所要時間に関する「乗り換えには何分ぐらいかかりますか？」といった発話や、難易度に関する「乗り換えは簡単ですか？」といった形式の発話が予測される。

その後、対話が展開されてシステムから s9 が入力されると、"MOP-道案内" の "場面-目的地" が活性化される。この場面には下位の MOP がないので、不足情報を収集するための発話についてのみ予測が行われる。この時予測される不足情報を収集するための発話は、活性化された場面と同じレベルに記述された場面から、所要時間に関する「出発点から目的地まで何分かかりますか？」といった発話や、上位の MOP に記述されている場面から、目印に関する「目印はありますか？」といった発話が予測される。

4. 音声出力のための対話管理

4.1. 焦点情報の抽出

概念表現からの音声合成において高品質の音声出力を行なうには、出力文における語順、助詞、表現方法、韻律的特徴、語句の省略・補充などに関して焦点の把握が重要な問題になる。また、入力も音声によって行なわれる機械との音声対話では次節でしめすように音声認識結果の提示と言う観点から、焦点情報を問題解決器からの発話に補うことも必要となる。ここでは、3.2.で述べたMOPを使って焦点を抽出し、補充することを検討する[5]。

4.1.1. 音声対話における焦点の補充

焦点は、対話の中で現在中心となっている概念で、文の中では省略されたり、代名詞化されることが多い。しかし、音声による機械との対話では焦点に関する語句を積極的に補うことも必要となる。例えば、次のような地理案内システムとの対話を考える。

U: 病院はどこですか？

S: この先の交差点の右にあります。

この対話では、システムの発話で「病院」に焦点があたっており、それが省略されている。このため、音声認識誤りのためユーザの発話が「美容院はどこですか」と誤認識され、「美容院」の場所が案内されていてもユーザは気がつかない。これを避けるためには、ユーザの発話の認識結果の確認するようなシステムの発話を挿入するか、あるいは焦点があたり省略されている情報を補充する必要がある。

4.1.2. MOP による焦点の抽出

MOP は場面の集合として表わされ、場面は同じ目標を持ち同時に起こり得る行動によって構成される。また、ある MOP のひとつの場面に対して、その場面を達成することを目標とする別の MOP が存在しえる。このような MOP を利用することによって、対話における場面の移行を把握することができる。対話管理において場面の移動を捉えるために、場面ごとにプランとその条件(ユーザあるいはシステム発話のテンプレート)を設定しておき、焦点があたることにより省略され得る情報を予め記述しておく。システム発話が充足したプランの条件部を利用して焦点を同定する。

4.1.3. 具体例

図4は地理案内システムにおける対話の流れを把握するために記述された MOP である。ここでは、"mop-案内"を最上位として、その下位に"mop-催し物会場"が用意されている。それぞれの MOP 中には数個の場面が設定され、場面ごとにゴールが与えられている。場面はプランを持っており、そのプランの発火する条件(図4では C1 から C10 で示されている。)が記述されている。発火条件中において、

mop-案内

場面:目的地決定(Goal:案内の目的地の決定)

Plan 目的地伝達 → mop-目的地確認

C1:教える ([Object(\$modify(?場所,?行き方))]).

C2:行く ([Sto(?場所))].

Plan 催し物伝達 → mop-催し物会場

C3:わかる ([Object(\$modify(#?催し物,?場所),\$negative]).

C4:教える ([Object(\$modify(#?催し物,?場所))]).

C5:行く ([Sto(?催し物))].

場面:目印決定(Goal:目印の決定)

Plan 目印伝達 → mop-目印確認

C6:\$is(#?目的地,\$place(?場所),[\$Sp-prominent(\$place)]).

場面:道案内(Goal:目的地までの道のりの伝達)

Plan

mop-催し物会場

場面:催し物会場決定(Goal:催し物の会場を決定する)

Plan 催し物会場質問応答 → SP-ASK-COMPONENT

C7:\$is(\$modify(#?催し物,?場所),どこ,[\$interrogative]).

C8:開く ([Object(#?催し物),\$place(どこ),[\$interrogative]).

Plan 催し物会場伝達

C9:\$is(?場所).

C10:開く ([Object(#?催し物),\$place(?場所)]).

C1-10: 対話と場面をマッチングさせる条件

?...:意味カテゴリー

#...:焦点となることにより省略され得る語句

図4 地理案内MOP

U1: 音声研究会へ行きたいので、場所を教えてください。

CU1-1: 行く ([Sto(音声研究会),\$wish]).

CU1-2: 教える ([Object(場所),\$request]).

CS1: \$is(会場,どこ,[\$interrogative]).

CS1': \$is(\$modify(\$focus(音声研究会),会場),どこ[\$interrogative]).

S1: 音声研究会の会場はどこですか。

U2: 機械振興会館です。

CU2: \$is(機械振興会館).

CS2: \$is(機械振興会館,\$place(\$modify(東京タワー,隣))).

CS2': \$is(機械振興会館,\$place(\$p-prominent(\$modify(東京タワー,隣)))).

S2: 機械振興会館は東京タワーの隣です。

図5 対話例 (2)

"?... " は意味カテゴリー、"#..." は焦点となることにより省略され得る意味カテゴリーを表す。図5は地理案内システムとユーザとの対話例を表す。S、U はそれぞれシステム、ユーザの発話文を表し、CS、CU はそれらの概念表現を表す。以下に、図5における図4の MOP 中の場面の移動および焦点抽出について述べる。

対話の開始時点における場面は、"mop-案内"である。まず、ユーザの最初の質問 U1 により概念表現 CU1-1,2 が生成される (音声理解システムによって、このような表現が生成されることを前提としている)。ここで、"#?催し物" と "音声研究会" を対応づけることで、"mop-案内" に含まれる C5 と CU-1、または、C4 と CU-2 とがマッチする。これにより、"plan-催し物伝達" が発火し、"mop-案内" から "mop-催し物会場" への移行が行われる。

次に、地理案内システムによって概念表現 CS1 が生成され、それが "mop-催し物会場" に含まれる C7 にマッチする。ここでは、"会場" と "?場所" とが対応づけられる。"#?催し物" とマッチする語句が CS1 に存在しないため、この語句に焦点が当たっていると判断できる。このため、ユーザへの出力文を生成する際には、音声認識結果をユーザに提示するために意味カテゴリー "?催し物" を持つ最新の対応語句である "音声研究会" という語句で補完するとともに、焦点が当たっていることを示す \$focus オペレータを付加する。この結果、対話管理部で修正された概念表現 CS1' が概念表現からの音声合成システムに送られ、出力文 S1 を得る。

ユーザの応答 U2 では主語が省略されているが、焦点が "音声研究会" であることより補完される。

次に、システムの応答の概念表現 CS2 とマッチする条件が "mop-催し物会場" に無い場合 "mop-案内" に戻り、C6 とマッチする。C6 においては、\$place が \$Sp-prominent により強調指定されているため、システムは発話文 S2 において「東京タワーの隣」を強調するためのオペレータを付加した概念表現 CS2' を音声合成システムに渡す。

4.2. 強調表現

強調すべき語句は次の三種類に分類できる[6]。第一に、対話のやりとりにおける文の表層的なパターンによって処理できる強調がある。この例として、

U: 地下鉄の池袋行きに乗るのですか？

S: 地下鉄の新宿行きに乗って下さい。

の対話例における「新宿行き」ように、システムの発話においてユーザの用いた語句の並びの一部が置き換えられたような場合が挙げられる。第二に、ユーザの質問に対するシステムの発話において回答の中心となる語句が強調されるべき語句となる。

U: 地下鉄をどこで降りるのですか？

S: 地下鉄は霞ヶ関で降りて下さい。

における「霞ヶ関」などである。第三に、問題解決器によって強調される語句が

ある。

U: 新宿からどういけばいいですか？

S: 地下鉄の銀座線に乗って下さい。

における「銀座線」がこの例である。「銀座線」を強調するためには、「新宿を通る地下鉄が複数ある」というドメイン知識が必要であるため、対話管理部ではこの判断を行なうことはできない。

4.3. 慣用テンプレートへの変換

概念からの音声合成システム SOCS では、よく用いられる表現に対して慣用テンプレートを用いることができる[1]。慣用テンプレートには韻律のパターンが記述されているため、慣用テンプレートを用いた方がうまく韻律的特徴を生成できる。このため、問題解決器によって生成された概念表現が SOCS で定義されている慣用テンプレートに変換可能であるときには、対話管理部でこの処理を行なう。例えば、

s1(開く ([\$Object(音声研究会), \$place(どこ), \$interrogative])).

は、

s2(\$where(開く ([\$Object(音声研究会)]))).

に変換できる。

5. むすび

本稿では、汎用的な音声インタフェースにおいて対話に固有の処理を行なう対話管理部に関して、音声入力と音声出力の両面からその処理内容について検討した。汎用的な音声インタフェースでは、問題解決器に依存しない処理方式でユーザ次発話の予測やシステム発話における強調の抽出を行なうが、効率の良い処理には問題解決器に依存した領域知識の利用も必要となる。汎用的なメカニズムと領域依存の知識を切りわけた枠組が汎用的な音声インタフェースには必要で、本稿で述べた対話管理では、問題解決器に依存した知識を MOP として利用している。今後、対話管理部を実現するとともに本手法の有効性を確認していきたい。

参考文献

- [1] 山下, 水谷, 溝口: 「合成音出力における概念表現の利用」, 電子情報通信学会技術研究報告, SP89-115, 41-48 (1990).
- [2] T.Yamamoto, Y.Ohta, Y.Yamashita, O.Kakusho and R.Mizoguchi: "MASCOTS: Dialog Management System for Speech Understanding System," Trans. of IEICE, Vol.E74, 7, 1881-1888 (1991).
- [3] R.C.Schank著, 黒川 他訳: 「ダイナミック・メモリー - 認知科学的アプローチ」, (株)近代科学社 (1988).

- [4] 吉田, 野村, 山下, 溝口: 「対話管理におけるMOPを用いたユーザ発話の予測」, 日本音響学会講演論文集, 2-P-22, 181-182 (1991.10).
- [5] 柴田, 山下, 溝口: 「概念表現からの音声合成における対話管理」, 日本音響学会講演論文集, 3-P-18, 327-328 (1991.10).
- [6] 山下, 柴田, 溝口: 「概念表現からの音声合成における文生成」, 日本音響学会講演論文集, 3-P-19, 329-330 (1991.10).

機械との音声対話におけるユーザ発話の予測について

山下洋一 溝口理一郎

(大阪大学 産業科学研究所)

1. まえがき

今日、社会生活におけるコンピュータの利用が増大し、情報の収集や検索、種々の企業活動や教育などにおいて計算機を利用して問題解決を行なうシステム(問題解決器と呼ぶ)の研究が広く行われている。現在、これら問題解決器とユーザとの意思伝達手段としては、主にボタンやキーボードを用いて入力が行なわれ、ディスプレイ上に表示される文字テキストによって出力が行なわれている。しかし、ユーザの負担などを考えるとキーボードや文字よりも音声によって情報伝達が行なわれる方が好ましい場合も多く、マンマシンインタフェースとしての音声情報処理技術の確立が期待される。本報告では、このような機械との音声対話における対話管理の役割、特にユーザ発話の予測について述べる。

2. 音声インタフェースにおける対話管理の役割

「音声による機械との対話」を人間同士の対話のように自然に行なうことは一つの大きな目標であるが、このような対話を実現するインタフェース技術を確認するためには、多くの問題を解決し

なければならない。この中で、対話に固有な処理として、語句の省略の同定・補充、利用者の意図の把握、意図を伝達するための最適な表現の決定などの問題を解決し、柔軟で高度な対話処理を行なわなければならない。さらに、音声理解を助けるための次発話の予測、避けることができないと思われる音声認識誤りや合成音声の聞き誤りへの対応、出力音声の韻律的特徴の生成など、自然言語など他の媒体には見られない音声特有の問題も解決しなければならない。これらの問題は対話あるいは音声対話に固有なものであり、問題解決器に依存するものではない。従って、機械との音声対話を行なう時、このような問題は個々の問題解決器内部ではなく、それとは独立したモジュールで対処すべきであると考えられる。このような観点から我々は、図1に示すような汎用的な音声インタフェースの構成を目指している。

図1に示す音声インタフェースでは、まずユーザの発話が音響処理された後で音声理解システムに入力される。音声理解システムは対話管理部からのユーザ発話の予測を利用しながらユーザ発話を同定し、結果を概念表現で記述し問題解決器へ送る。問題解決器は、処理(推論)結果を概念表現で記述し対話管理部へ戻す。対話管理部は対話コンテキストに従って情報を概念表現に付け加え、

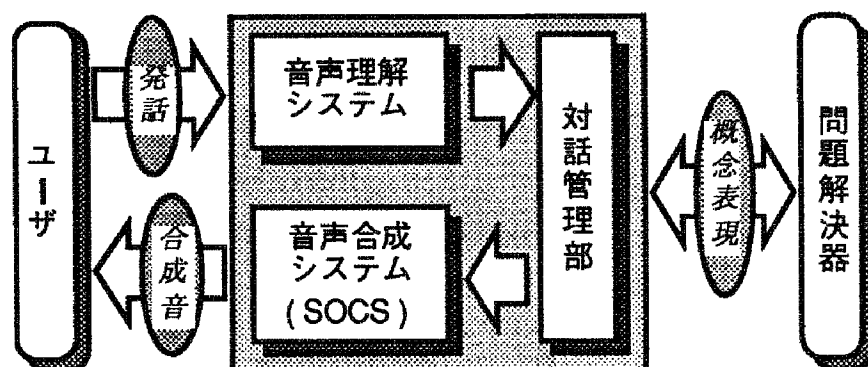


図1 音声インタフェース

それを概念からの音声合成システムSOCS[1]へ渡す。さらに、概念表現からの音声合成システムが音声を出力する。このような枠組における対話管理は、音声入力に関する処理と音声出力に関する処理に大きく分けることができる。これを簡単にまとめてみると、

(1) ユーザから問題解決器への音声入力に関する処理

(1-1) ユーザ次発話の予測

音声理解システムを支援するために、対話コンテキストからユーザ発話を予測する。これには、対話が要求と応答のペアから構成されるという対話に関する知識や、問題解決器においてどのような話題が現われるかといったドメイン知識が利用できる。

(1-2) 問題解決器への情報の標準化

一般に、ユーザは同じ意味に対していくつかの異なる表現をとることができる。そこで、対話管理部がユーザの意図を判定し、一定の表現で問題解決器に情報を渡す。

(2) 問題解決器からユーザへの音声出力に関する処理

(2-1) 焦点情報の抽出

一般に人対人の対話では焦点のあたっている情報が省略されることが多いが、機械との音声対話では音声認識結果の提示と言う観点から、焦点情報を問題解決器からの発話に補うことも必要となる。また、焦点は概念表現からの音声合成にも重要な情報となる。

(2-2) 新情報の抽出

対話において新しく現れた情報を対話管理で管理し、音声合成システムで利用する。合成音の聴取では、始めて聞く言葉(単語)が聞き取りにくいことがあるので、例えば、発話速度をやや遅くするなどの処理が必要になる。

(2-3) 強調表現

問題解決器の発話において強調される語句の中には、対話のコンテキストから強調されるものが多い。対話管理部では、このような語句を抽出し音声合成システムに渡す。

(2-4) 慣用テンプレートへの変換

概念表現からの音声合成では、よく用いられる表現に対して韻律パターンを設定した慣用テンプレートを準備しておくことができる。問題解

決器から入力された概念表現を慣用テンプレートをを用いた表現に変換することにより、より高品質な合成音を出力することが可能となる。

となる。本報告では、(1-1)の「ユーザ次発話の予測」について以下で述べる。

3. ユーザ発話の予測

3.1. SRプランに基づいたユーザ発話の予測

我々はこれまでに、MASCOTS と呼ばれる対話管理システムを構築し、SRプランに基づいた対話の理解とユーザ発話の予測について検討してきた[2]。そこでは、対話が主に要求と応答のやり取りから構成されることに着目して、このやり取りを17個のSRプランに分類した。17個のSRプランは要求を行なう対話者が問題解決器かユーザかで、それぞれシステムSRプランとユーザSRプランに分類され、実行時に活性化されたプランはシステム用とユーザ用の2本のスタックによって管理される。この枠組におけるユーザの次発話の予測は、プランの同定と密接に結び付いて行なわれる。ユーザが応答を返す場合には問題解決器の要求の発話によってすでに一つのシステムSRプランが活性化されており、このSRプランによってユーザの応答が予測される。ユーザが要求を出す場合には、音声理解システムに入力された文節ラティスの情報からボトムアップにどのユーザSRプランが活性化されたかを決定する。また、個々のSRプランにはユーザ発話を問題解決器に渡す時に取るべき表現が記述されており、問題解決器への情報の標準化を行なうことができる。

3.2. ドメイン知識を利用したユーザ発話の予測

ユーザ発話を有効に行なうために、ドメインに知識として話題に関する情報をあらかじめ用意しておくことができる[3]。あるドメインにおいて対話を行なう場合には、話題の遷移をある程度限定でき、ユーザ発話に現われるキーワード等がうまく予測できる。従来のMASCOTSでは、要求と応答の組に基づいた対話の局所的な情報を用いてユーザの次発話を予測していた。しかし、ユーザの要求などのように局所的な情報だけでは十分に予測できない発話も存在する。そこで、対話の領域に依存する知識として話題の遷移に関する情報を利用しユーザ発話の予測を行なう[4]。

情報のやり取りに直接関与するユーザの発話は 1) 応答、2) 情報の詳細化を行うための要求、3) 不足情報を収集するための要求、4) 評価、5) 確認、6) 話題の更新の計6種類に分類できる。一般に対話では、ユーザの次発話としてこれらのどの種類の発話も行うことが出来るため、次発話を予測するには全ての発話の種類に関して発話文の構文や主題を予測しなければならない。ここで、返答の発話形式は直前の要求型の発話の構文や意味内容に強く依存するという対話の局所的な性質から、1)、4)、5)のユーザの発話は、システムの発話からある程度予測することができる[2]。しかし、2)、3)、6)の発話は、領域に依存する知識がなければ十分な予測を行うことができない。

3.2.1. 話題パッケージネットワーク：TPN

ある特定のドメインで対話が行なわれる時には、対話に現われる話題はそのドメインごとに限られたもので、次にどのような話題に遷移するかもそこまでに現われた話題からある程度推測することができる。そこで、話題の遷移に関する情報を話題パッケージネットワーク (TPN: Topic Packet Network) としてあらかじめ対話管理部に与えておく。図2に、地理案内を行なう際に用いられる TPN の例を示す。

図2の TPN は、「地理案内」という話題パッケージが、まず「ユーザの目的地」あるいは「参加する催しもの」が話題になることを示している。さらに、話題「目的地」からは「目的地案内」パッケージ中の「出発点」「目印」「交通機関」「徒歩」

「徒歩」「所要時間」の話題に遷移することができる。一つ的话题が終了した場合には、それと同じパッケージ内の話題への遷移や、元の話題への復帰が起こる。一般に一つ的话题は、さらに詳細な話題あるいは連続して起こる話題へと遷移する。

3.2.2. TPN を用いた発話予測

対話管理部はシステムの発話を受け取り場面の同定を行った後、上述したユーザの発話分類に基づいて発話予測を行う。従来から用いているSRプランによって1) 応答、4) 評価、5) 確認を予測し、さらに、ドメイン知識に依存した予測として、2) 情報の詳細化を行う要求、3) 不足情報を収集するための要求、6) 話題の更新の予測を行う。

情報の詳細化を行う要求、および話題の更新は、ユーザが現在活性化している話題に関してより詳細な情報を要求する場合に行われる。従って、ユーザの次発話の話題は、その時活性化されている話題に接続する話題パッケージへ移動すると考えることができる。

一方、不足情報の収集を行う要求は、その時活性化している話題に関してユーザが十分な情報を受け取った後、別の話題に関する情報を要求する場合に行われる。従って、ユーザの次発話の話題は、その時活性化している話題を含むパッケージ中の別の話題へと遷移する。

従来、SR-プランによって生成されたキーワードが音声理解システム ASP へ送られて文節候補ラティスの解析が行なわれていた。TPN を合わせて用いた予測では、各話題にあらかじめキーワード

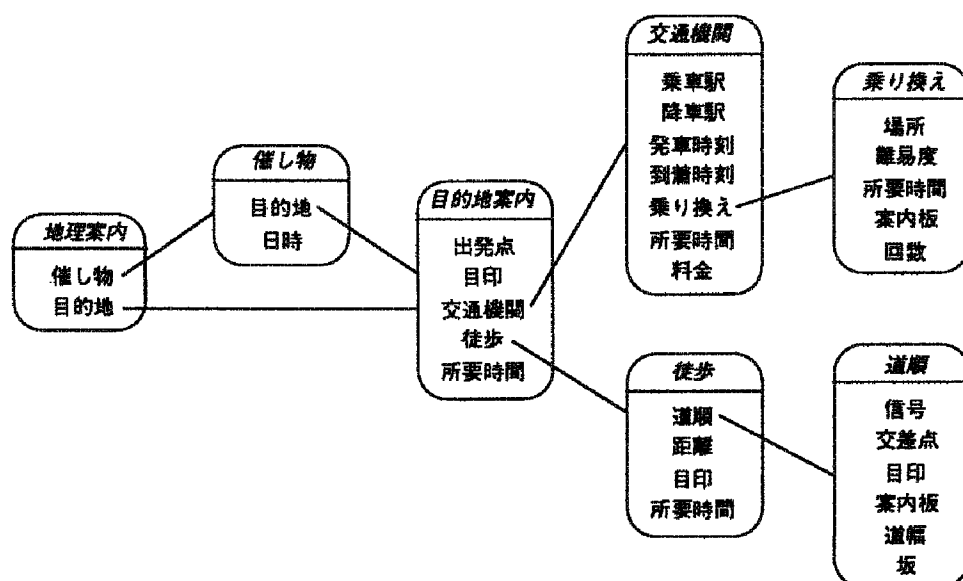


図2 話題パッケージネットワーク

U1: 音声研究会へ行きたいのですが、どのように行けばよいか教えてください。
 S1: 音声研究会の会場はどこですか。
 U2: 機械振興会館です。
 S2: JR東京駅まで新幹線で行ってください。
 S3: 東京駅で地下鉄丸ノ内線に乗り換えて下さい。
 U4: どこで降りるのですか。
 S4: 銀座で降りて下さい。

(S: システム、U: ユーザ)

図3 対話例

を与えておき、遷移可能な話題に関連したキーワードが SP-プランによるキーワードと合わせて ASP に送られる。

3.2.3. 動作例

図2の TPN を用いて、ユーザと地理案内システムの対話 (図3に示す) におけるユーザ発話 u4 を処理する際の発話予測について述べる。

まず、システム側の発話 s3 まで処理が進んだ段階で、話題は「電車移動」パケット中の「乗り換え」となっている。ここで、次のユーザ発話では、情報の詳細化を行なうために次の話題パケットへ遷移した「所要時間」や、不足情報を補うためにパケット中の他の話題に遷移した「降車駅」などの話題へ遷移可能である。そこで、このような話題に関連するキーワード「何分」「かかる」および「どこで」「降りる」などが選択され、SR-プランによって生成されるキーワードと合わせて ASP へ送られる。ASP では文節ラティスを解析し、SR-プランと話題のそれぞれの候補に得点づけを行ない、それに基づいて SR-プランと話題を選択する。この時、「どこで」は SR-プラン UP-ASK-COMPONENT によっても生成されるキーワードで、このような SP-プランと TPN の両者によって生成されるキーワードが文節ラティスに存在すれば、高い得点を与える。

4. むすび

本稿では、汎用的な音声インタフェースにおいて対話に固有の処理を行なう対話管理部に関して、ユーザ発話の予測について述べた。今後、ユ

ーザ発話予測における話題の利用方法をさらに検討するとともに、本方式の有効性を確認して行きたい。

参考文献

- [1] 山下, 水谷, 溝口: 「合成音出力における概念表現の利用」, 電子情報通信学会技術研究報告, SP89-115, 41-48 (1990).
- [2] T.Yamamoto, Y.Ohta, Y.Yamashita, O.Kakusho and R.Mizoguchi: "MASCOTS: Dialog Management System for Speech Understanding System," Trans. of IEICE, Vol.E74, 7, 1881-1888 (1991).
- [3] 新美, 小林: 「対話音声理解システムにおける話題管理とキーワード予測」, 電子情報通信学会技術研究報告, SP87-103, 37-42 (1988).
- [4] 吉田, 野村, 山下, 溝口: 「対話管理におけるMOPを用いたユーザ発話の予測」, 日本音響学会講演論文集, 2-P-22, 181-182 (1991.10).

日本語音声応答システム

－ 特にその対話管理について －

広瀬 啓吉 浅野 康治

(東京大学 工学部)

1. はじめに

1-1. 本研究の目的

コンピュータを用いた知的システムと人間とのインターフェースを考えると、これまで実現されているシステムにおいては、ユーザ（人間）はシステムの恩恵に浴する前に、そのシステムとのインターフェースになれる必要があり、この点で従来のシステムはユーザに苦痛を強いるものが少なくない。このようなユーザ側にとっての苦痛を軽減し、マンマシンインターフェースを改善するのに最も効果的な方法の1つは、情報伝達の主な手段である自然言語、そのなかでも特に人間に対する負担の小さい音声言語をコンピュータと人間の間の情報伝達に用いることであろう。

本研究においては、ユーザである人間との間で双方向に情報伝達を行なうシステムの1つとして質問・応答システムを取り上げ、ユーザとの間の対話を通じて、ユーザが必要とする情報を適切な自然言語で表現し、連続音声の形でユーザに提示するシステムを構築することを目的とする。

1-2. 本研究の範囲

1-1. で述べたように、本研究は日本語による音声応答システムの作成を目的とする。当然のことであるが、システムからユーザへの応答に音声言語を用いるのであるから、ユーザからシステムへの入力にも音声言語を用い、音声による双方向情報伝達を行なうシステムを構築するのが理想である。しかし、それには連続音声認識・理解という困難な問題を解決する必要がある。この問題に関しては現在も様々な研究機関で様々な形で研究が進められているので、本研究では対象外とし、研究範囲としてはユーザから与えられた入力によって得られる情報に基づいて適切な応答文を生成し、それを連続音声の形でユーザに提示するシステムの開発に限定することとした。そのためシステムへの入力としては、ユーザからの入力音声に音声理解処理を施した結果として得られるであろう深層意味表現を用いることとした。

しかし単なるQAシステムのようにユーザからの質問にシステムが単純に答えるというようなシステムではなく、必要に応じてシステムはユーザに対してinterruptiveに質問や提案を行ない、お互いが主導権を取り合う対話にも対処できるよ

うなシステムの構築を目指している。これによってシステムは必要な情報を収集し、ユーザにとってより適切な情報を提供できるようになり、よりUser friendlyなシステムを構築することができると考える。

本報では、以上で述べた日本語による音声応答システムについて、そのシステム全体の構成と、その重要な機能の1つである対話管理に関してこれまでに行ってきた研究内容について紹介する。

2. システム全体の構成

システムの全体構成の概略を図1に示す。応答システムを構成するにあたっては、当然のことながら質問応答の話題となる分野に関する知識を提供できるエキスパートシステムやデータベースなどの知識ベースが必要となる。本研究においては、知識ベースとして次のような理由から一般に提供されているスキー場に関するデータベースを用い、質問応答の内容としてはスキー場の案内及び紹介とした。

- ・対象分野に関する対話において用いられる語彙の量が数百個程度ですむので、単語辞書の作成にかかる労力を抑えられる。
- ・データベースがテキストデータ、数値データ、二値データをそれぞれ含んでおり、データの種類のバラエティに富む。
- ・本システム専用にデータベースを作るのではなく、一般に提供されているものを用いることによって、データベースの種類によってシステムに変更を要する部分を明確にし、さらにその変更のプロセスを明らかにすることによって、他のデータベースに対してもこのシステムの構成やアルゴリズムが有効であることを示す。

各処理部の内容は節を改めて説明するとして、ここでは処理の過程で参照されるデータを保存する知識ベースについて概説する。

データベース (DB) :

前記の通り本研究においてはデータベースとしてスキー場に関するものを用いることとした。これには各スキー場に関してリフトの数や各種料金、現地までの交通手段などのデータが登録されている。

単語辞書:

話題によって、対話において出現する単語は当然変わってくる。そこでシステムにおいて利用するデータベースの種類に応じて、使用する語彙を登録した単語辞書を用意する必要がある。辞書中において登録されるべき情報は単語の品詞によって違い、以下のようにまとめることができる。

- ・全ての単語において登録される情報
単語表記、品詞、音素表記、アクセント型
- ・動詞、形容詞などの述語において追加して登録する情報

深層格、表層格、必須・選択格の別

データベースの検索条件との対応関係

・名詞において追加して登録する情報

動詞、形容詞との共起情報

データベースのフィールド名との対応

対話管理ルールベース：

システムとユーザの間の対話を表現し、管理するための規則が記述されているルールベースで、詳細は対話管理処理の節で説明する。

対話履歴ベース：

ユーザとの間で円滑に対話を行うためには、システムは現時点までの対話の履歴を保存して、焦点や省略・照応表現の処理に利用するのはもちろん、ユーザに対する伝達情報をプランニングする際にもこれを参照する必要がある。対話履歴ベースはこれらの情報を蓄積するためのもので、詳細は対話管理処理の節で説明する。

文法ルールベース：

応答文を生成する際に用いる文法規則を記述したルールベースである。本システムは深層意味表現として格構造を用いるため、文法規則としては主として語順に関する規則になる。

音韻ルールベース：

連濁など単語辞書中の音素表記とは異なった読みとなる異音化処理を行ったり、単語のアクセント型の変更、休止時間長やイントネーションの制御などの韻律制御を行う際の規則が記述されているルールベースで、音韻処理において利用される。

音声合成器制御用パラメータベース：

規則合成のために合成器の制御パラメータの時間パターンを蓄積してあるのがこの音声合成器制御用パラメータベースで、本システムでは蓄積の単位として日本語CV音節を採用している。

3. 深層意味表現

この節では本システムにおいて用いる深層意味表現について、その具体的な定義を行なうこととする。

深層意味表現とは、表層文において表現されている意味内容に関する情報だけを抽出して表現したもので、統語情報などは含まれない。この深層意味表現には一階述語論理や R. C. Shank による CD 表現などいくつかの表現方法が提案されているが、本システムにおいては以下の理由から C. Fillmore によって提案された格構

造の概念を用いることとした。

・システムの目的上、対象となる言語は日本語だけであるので、中間言語表現を用いた機械翻訳システムなどのように言語間の差異を吸収するために深層意味表現としてかなり抽象的なレベルのものを採用する必要がない。

・本システムへの入力としてとりあえず深層意味表現を用いるので、人間が実際に入力したい表層文から深層意味表現をある程度容易に導出できる必要がある。

具体的な格構造としては、次のように記述することとした。

(述語 (意味付加) (文脈情報)

((深層格 格のfiller) (深層格 格のfiller) ...))

この表現のうち、意味付加とは時制、態、丁寧など述語の活用と助動詞の付加によって表現できる意味情報を表わし、また文脈情報とは複数の格構造間の意味的接続や対話状況におけるこの格構造の果たす役割などに関する記述である（理由、前提条件の設定など）。さらに対話状況から焦点がおかれたり、表層表現において省略・照応表現を用いるべき深層格にはマークを付与する。

この格構造において用いる深層格としては、一般性を持たせるために機械翻訳システムの開発を目指したMuプロジェクトにおいて用いられている深層格の種類を参考にした⁽¹⁾。Muプロジェクトにおいては動詞・名詞間の深層格として全部で34個が用意されているが、本システムにおいては、このうち対象となる分野の対話を表現するのに必要であると考えられる14個を用いることとした。本システムにおいて用いる深層格の一覧を表1に示す。

最後に例として、深層意味表現とこれを意味ネットワーク形式で表現したもの、及びこれによって表現される表層文の1例を図2に示す。

4. 対話管理部

対話管理部ではユーザとシステムの間に対話履歴の管理と、この履歴情報をもとにユーザからの入力に対してシステムが応答する内容を深層意味表現の形で生成する処理が行なわれる。このためには、まず対話の流れを表現する方法を考え、さらにその表現に基づいて対話を管理し、応答内容を立案する処理を実現する必要がある。本節ではこれらの処理を実現するために、まず質問応答対話の一般的な対話の流れとその表現方法を検討し、さらにその結果に基づく対話管理・応答内容立案処理の概略について説明する。

4-1. 対話の流れとその表現方法

質問応答対話は、通常対話当事者双方による次の2種類の発話が1組となって1つのルーチンを形成し、これが組み合わされて全体の対話が構成されていると考えることができる。以下では、これを対話の基本ルーチンと呼ぶ。

①質問する主題の設定及び質問

②質問に対する応答

また、音声による対話では文字による対話よりもはるかに容易に相手の発話に対して割り込めること、音声対話における個々の発話は一般に短いことなどから、対話の途中で割り込みが数多く生じる。この割り込みは協調的なものと競合的なものの2種類に分類することができる。

協調的な割り込みとは、相手の発話内容の一部が理解できずに質問する場合など、現在の対話の流れに沿った上での割り込みで、この場合は割り込みで要求された処理を実行した後に、対話の流れをもとの状態に戻すことができる。また、この協調的割り込みが生起する割り込みの大半を占める。一方、競合的な割り込みとは、一方の発話目的が達成されないうちに、他方から現在の対話の流れに無関係な別の主題に関して発話が行われるというものである。

対話管理部でこれらの対話の流れを表現する方法として、対話の基本ルーチンをもとに対話の流れを表現することが考えられる。すなわち、①の主題の設定及び質問の発話によって基本ルーチンが導入され、②の応答によって基本ルーチンは終了し、①と②の発話の対応関係が完結する。そして協調的割り込みはこの基本ルーチンの入れ子で表現し、内部の入れ子のルーチンが終了するとその外側のルーチンに対話の流れが戻るようにする。基本ルーチンが入れ子構造となる対話の簡単な例を図3に示す。また、1つの発話によって複数の基本ルーチンが終了する場合も有り得る。

4-2. 対話管理処理

対話の基本ルーチンに基づいて対話の表現を行なうとすると、対話管理処理として行なうことは基本ルーチンの導入、終了のルーチンの管理である。このルーチン管理には、対話の話題に依存する規則も当然必要となるが、本システムを構成するに当たっては、使用するデータベースが変更されてもシステム構成やアルゴリズムが有効であることを示すという目的があるので、対話管理に関する規則を話題に依存せず質問応答対話一般について成り立つ規則と、話題やデータベース検索というタスクに依存する規則に分け、データベースの変更などに伴う変更部分をなるべく少なくかつ明確化するように配慮している。

具体的には、対話管理に関する規則を以下の3つに分類し、データベースの内容の変更による話題の変更やタスクの変更による対話管理規則の更新部分を局在化かつ明確化する事を目指している。

・質問応答対話一般に関する規則

基本ルーチンの導入・終了の制御を直接司る規則

・話題に依存する規則

話題に固有の対話の進め方のプランを対話の状況ごとに記述した規則

・タスクに依存する規則

データベース検索というタスクに固有の対話の進め方のプランを記述した規則

4-2-1. 一般質問応答対話規則

まず質問応答対話一般に関する規則として、本システムではユーザからの入力を図4のように分類し、この分類ごとに基本ルーチンの管理を行う。

① 質問系のユーザ入力時

システムは新たに基本ルーチンを導入するとともに、質問内容に基づきデータベース検索を行う。検索結果として質問に対する回答が得られた場合はその深層意味表現を生成して基本ルーチンを終了し、1つ外側の基本ルーチンに戻る。一方、回答が得られない場合はその状況に応じて話題依存・タスク依存の規則を適応してユーザに対して質問や要求などの協調的割り込みを行う。この場合は発話内容に関する深層意味表現を生成するとともに新たな基本ルーチンを導入する。

② 要求系のユーザ入力時

基本的には①の質問系の処理の場合と似ており、要求系の入力によって基本ルーチンを導入、その内容を処理した結果、要求に対する確認、受諾、要求内容の実行が得られた場合に、応答内容の深層意味表現を生成するとともに、基本ルーチンを終了し1つ外側の基本ルーチンに戻る。

③ 応答系のユーザ入力時

この入力が許容されるのは、いちばん内側の（現在処理対象となっている）基本ルーチンがシステムの発話によって導入され、しかも終了していないときであり、その場合以外は不適切入力としてその旨をユーザに伝達する。条件を満たす場合はその内容がシステムの発話に対する応答として適切であるか（システムが要求した情報と深層格、意味素性などが一致するか）を判断し、適切であればこの基本ルーチンを終了させ、1つ外側の基本ルーチンに戻る。また回答として不適切な場合はやはり不適切発話として扱う。

④ 外側の基本ルーチンに戻ったときの処理

①から③の処理のなかで基本ルーチンが終了し1つ外側の基本ルーチンに戻ったときの処理は、次の2つの場合に分けて行われる。

・戻ってきた基本ルーチンがユーザの発話によって導入されたとき

システムはその基本ルーチンが導入されたときのユーザ発話の内容に従って、応答内容を立案する。

・システムの発話によって導入されたとき

システムはユーザからの入力待ちの状態になるが、しばらく待って入力がない場合は、システムからユーザに対して入力を催促する発話を行う。

⑤ その他

ユーザの発話中に知らない語彙がある場合に、それについて質問するなど、人

間同士の対話において一般に行われる言語行動をプランの形で規則として記述しておき、必要に応じて適用する。

以上の規則に従って基本ルーチンの導入、終了を行うことによって対話の管理を行うことができる。しかしこれは協調的割り込みに関しては考慮されているが、競合的割り込みについてはまだ十分に考慮されていない（不適切発話も一種の競合的割り込みと考えられる）。競合的割り込みはシステム側からは発生を陽に抑制することができるが、ユーザ側からは頻度は非常に少ないが発生する可能性がある。この対処法としては、ユーザに割り込み発生前後のどちらの主題を優先させるか問い合わせる方法、前の主題を捨て去る方法、新たな主題に関する処理の終了後にシステムが前の話題に戻す方法などが考えられるが、どの様な場合にどれが最善であるかはまだ検討段階である。

4-2-2. 話題依存規則とタスク依存規則

一般質問応答対話規則で対話の基本ルーチンの導入、終了の処理が行われるが、ここでは話題に依存しない規則だけが記述されているので、これだけで対話を管理することは不可能である。そこで本システムでは話題依存規則とタスク依存規則の2つの規則を別に設けている。

話題依存規則はユーザとシステムの間で行われる対話の話題、すなわちデータベースの内容に依存する規則である。具体的には、ユーザの質問内容が漠然としている場合（例：「今度スキーに行きたいのですが、どこがいいですか」）などでデータベース検索ができず、ユーザの質問・要求に回答できない場合に対処するため、状況に応じて対話をどの様に進めて行くかのプランを記述しておき、それによってユーザへの応答内容を決定する。

タスク依存規則はこのシステムがユーザとの対話を通じてどの様なタスクを実行するかという点に依存する規則である。すなわち本システムの場合はデータベース検索を行うことを想定しているので、データベース検索の結果、該当するものが多すぎてユーザにすべて提示しきれない場合に、検索条件の追加を要求するなどのプランを記述しておく。

4-3. データベース検索とシステム発話内容の決定

具体的なデータベース検索処理はデータベースの構造に依存するので、データベースとのインターフェースもデータベースに依存して変更を加える必要がある。この処理では、基本ルーチンによる対話の管理の結果としてシステムがデータベース検索を行う必要が生じたとき、ユーザ入力をデータベース検索に適した形式に変換し、データベースへのアクセスが行われる。変換はユーザ入力中に含まれる単語（概念）とデータベースの項目や検索条件との対応を単語辞書中に記述しておき、それに基づいて行われる

システムの発話内容は、図4のユーザ入力のカテゴリごとに規則化し、それとデータベースの検索によって得られたデータを用いて決定される。また4-2-2.で述べたように、話題依存・タスク依存の規則によってもユーザに対する質問・要求系の発話内容が生成される。これについての詳細は紙面の関係で割愛する。

4-4. 対話履歴ベース

対話履歴ベースとしては、ユーザとシステムのそれぞれの発話を深層意味表現の形で表現したものと、対話の基本ルーチンによる発話間の対応関係を保存することによって実現する。さらに省略・照応表現や焦点の移動の制御に用いるために、各深層格の最も新しい時点でのfillerの内容と、現在焦点がおかれている深層格を保存しておく必要がある。

5. おわりに

筆者らが現在進めている日本語による音声応答システムの実現を目指した研究について、システム全体の概要及び対話管理処理について概説した。これまであまり研究の進んでいない人間の音声による対話活動に関して、本システムは、それを構成する過程においていくつかの知見や仮説を得ることを目指し、さらに音声対話に固有の音声現象、特に韻律的特徴に関する分析を進め、それらの知見をシステムに組み込むことによって、将来的によりUser friendlyなシステムとして発展する可能性を秘めていると考えている。

参考文献

- (1) 辻井潤一：“機械翻訳システムにおける辞書の構成，”野村浩郷編“言語処理と機械翻訳”，pp.71-105（1991）。

表1 本システムにおいて用いる深層格

主体 (Subject)	時・終点 (Time-To)	場所・経過 (Space-Through)
対象 (Object)	時間 (Duration)	原因・理由 (Cause)
相手1 (Partner)	場所 (Space)	手段・道具 (Tool)
時 (Time)	場所・始点 (Space-From)	目的 (Purpose)
時・始点 (Time-From)	場所・終点 (Space-To)	

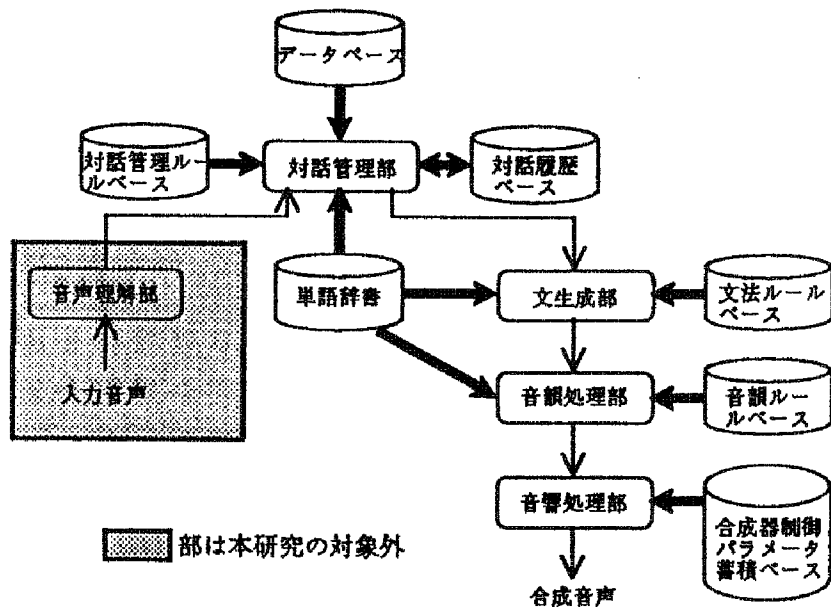
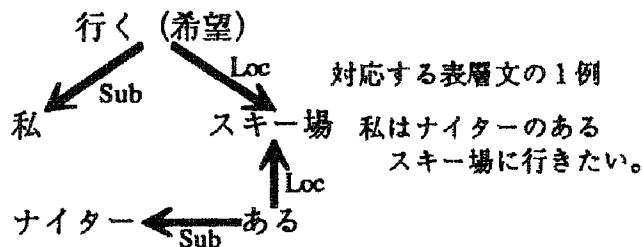


図1 システム全体の概略

(行く (希望) (ϕ)
 ((主体 私) (場所 スキー場[1])))
 (ある (ϕ) (限定)
 ((主体 ナイター) (場所 スキー場[1])))
 [1]は同じものを指していることを示す添字

リスト形式による格構造表現



ネットワーク形式による表現

図2 深層意味表現の例

U: ○○スキー場へは何時間かかりますか。
 (設定、質問)
 S: 車で行きますか、JRですか。
 (設定、質問)
 U: 車で行きます。 (応答) (1)
 S: それでは4時間くらいです。 (応答) (2)

(1) ユーザ (U) によるルーチンの導入
 (2) システム (S) によるルーチンの導入

図3 基本ルーチンの入れ子構造の例

- ・ 質問系のユーザ入力
 - 深層格のfillerに関する質問
 - Yes-No型の質問
 - Which型の質問
- ・ 要求系のユーザ入力
- ・ 応答系のユーザ入力
 - 質問に対する回答
 - 要求に対する受諾

図4 ユーザ入力の分類

音声対話のいくつかの言語的な特徴と対話システム

中川聖一 山本幹雄 小林聡 田口勝豊

(豊橋技術科学大学)

1. はじめに

自然言語による対話は、次世代のマン・マシンインターフェースとして最も注目されているものの1つである。しかし、これまで開発された自然言語対話システムの多くはタイプ入力を前提としており、一部には有用であっても大半の場合はコマンド入力やメニュー方式に勝るとは思われたい。その特性を十分発揮するためには、タイプ入力ではなく音声入力である必要がある。音声入力による自然言語インターフェースは人間にとって自然であり、前もっての訓練、特殊な技能などが必要でないため、計算機になれていない一般ユーザとのインターフェースとして特に大きな意義を持っている。

現在の音声言語処理技術でも孤立発声された単語や丁寧に発声された連続音声の認識は実用段階を迎えている。しかし、自然に発声された連続音声の認識、自然な対話を行う技術などに問題点がある。

本研究では、音声言語の言語的特徴、音声対話の構造的特徴、より自然な対話を行うための基礎的な検討、および対話システムのプロトタイプの試作を行った。

2. 音声対話の言語的特徴 [1] [2] [3]

音声対話システムを構築するには、音声言語の統語・意味・語用論的特徴のデータが不可欠である。これまでの自然言語処理の研究では、その多くが書き言葉を対象としているが、音声による発話文には一般に、書き言葉と異なる点が多く、書き言葉のデータをそのまま利用するわけには行かない。最近、実際の音声による発話を対象とした分析も盛んに行われているが、まだ解決すべき多数の問題が残っている [1] [2] [3] [4] [5]。

本章では書き言葉に対する文の解析手法を音声発話文に適用する際、とりあえず問題となると考えられる名詞の付属語の脱落と文節の倒置の特徴分析の結果を述べる。

2.1 使用した対話データベース

音声対話分析用として、日本音響学会連続音声データベースのシミュレーション対話の書き起こしテキストの一部を使用した。また、キーボード対話の分析に

は我々の研究室で収録したものを使用した。対話のドメインは音声とキーボード対話どちらも観光案内など各種相談である。それぞれの現象の分析で対象とした文の集合は以下の通りである。

(1) 付属語が脱落している文節の分析対象

音声対話のみを使用した。対話数：19対話、総文数：1769文、文節数：12856（間投詞、相づち、言い直しを省いた文節数は8215）。

名詞の助詞の脱落・誤りを分析する場合の問題点は、音声をテキストにする段階で無意識の内に助詞を入れてしまったり、誤りを正してしまう可能性である。我々は19対話1769文に対して、テキストの元となった音声テープと、書き起こされたテキストとを3回照らし合わせて誤りをチェックした。

(2) 倒置文節の分析対象

音声対話のみを使用した。倒置文節の分析では、完全に音声と一致がとれていない書き起こしテキストを使っても大きな問題は起きないため、音声テープとの対応のチェックを行っていない他のテキストも分析に加え、結果的に約3000文を分析対象とした。

(3) 質問応答対のネストの分析対象

キーボード対話は、6名による8対話、全部で約500文を使用した。音声対話は、36対話、全部で約3000文を使用した。

2.2 名詞の付属語の脱落^{〔1〕〔2〕}

係り受けに基づく構文解析および日本語の意味解析では名詞文節の助詞によって表現されている格情報が重要な役割を果たしている。助詞の省略は、格情報の欠落を意味するため、構文・意味解析にとって大きな問題となる。本節では、実際の音声対話文での名詞の付属語の脱落を分析する。

2.2.1 係り先までの距離

表1は名詞文節全体、および付属語が脱落した文節に対する、係り先までの距離をまとめたものである。ここで、距離とは係り先の文節までの文節の数である。距離の欄の数字は係り先までの文節の数を表す。例えば、直後の文節に係る文節は距離1の文節に係る。「なし」の欄は係り先がない文節を表す。各付属語に対する数字は「／」の左側が付属語脱落文節のもの、右側は名詞文節全体（付属語脱落文節を含む）の数である。脱落した付属語は文脈を考慮に入れて我々が推定した。省略に対する分類で注意する点は以下の通りである。

- (1) 省略された「は」については、「が、を」などの格を持つ場合は格助詞として分類した。主題としての「は」と「～は～です」の「は」だけが、「は」として扱った。よって「排他的な「は」の使用」などは格助詞として扱っている。
- (2) 省略における「へ」は「に」に統一した。
- (3) 「1文節文」とは付属語のない名詞1つからなる文である。「1文節文」は非

常に特殊なので以下の検討では考慮に入れない。また、「です」は「そういった訳。」における「訳」の後の「です」の省略である。多くは文末に現れる。

表1 付属語脱落文節に対する係り先の距離

	1	2	3	4以上	なし	合計
が	36/271	3/65	2/20	1/11	0/11	42/378
を	30/166	7/24	1/10	1/6	0/4	39/210
に	10/250	3/57	2/23	0/34	0/11	15/375
へ	--/9	-/2	-/1	-/0	-/0	-/12
で	2/9	3/44	1/17	0/38	0/15	6/267
まで	0/20	0/9	0/3	0/3	0/1	0/36
から	0/42	0/23	0/12	0/7	0/8	0/92
は(主題)	0/--	7/--	2/--	2/--	0/--	11/--
は(~です)	8/--	2/--	1/--	0/--	0/--	11/--
は全体	8/117	9/87	3/61	2/58	0/23	22/346
の	3/613	0/25	0/7	0/5	0/16	3/666
と(並列)	13/165	0/23	0/15	0/41	0/12	13/256
その他	0/318	0/80	0/56	0/76	0/90	0/620
1文節文	--	--	--	--	7/7	7/7
です	--/23	--/6	--/22	--/42	5/360	5/453
合計	102/2147	25/445	9/247	4/321	12/558	152/3718

・脱落の具体例(括弧の中のローマ字が省略された付属語)

が(動作主など): 高速(ga)できてますので...

を(対象など): 馬(wo)飼ってますんで...

に・へ(場所など): アメリカ(ni)行ってから

で(道具・場所など): 100000弱(de)往復は買える。

は(主題): これ(ha)、もともと車の値段が安いもんですから、

は(です): こちら(ha)豊橋技科大です。

の: 九州(no)福岡

と(並列): テニス、ゴルフがあります。

助詞によって脱落の頻度が異なることが明かである。「が」と「を」が際だって脱落しやすい助詞と言える。また、「が、を、に、で」の必須格になりやすい格助詞だけで全体の約70%、さらに「は」を加えると約85%を占める。「まで、から」などの任意格である格助詞は脱落していない。

助詞落ちの文節は、直後にかかる可能性が最も大きく、全体の67%を占める。2つ先の文節までだと、84%になる。また、副詞を飛び越えて2つ先の文節に係る文節は「直後」に係ると言うことにすれば、直後に係る文節は76%を占める。

述部に係る格助詞が省略された文節を対象に、係り先の述部までの述部の数を分析したところ、最も近い述部に係る場合が97%であることが分かった。また、2つ目以降の述部に係る例は3例だけであるが、そのうち2つはそれぞれ、慣用的に2つの文節を1つにまとめれば1つ目の文節といえる（例：「見て回る」）か、または、言い直しに近い文である。ただ1つの例が入れ子になっている単文をまたいでいる。

2.2.2 係り先の述部の必須格と脱落した助詞の関係

述部に係っている文節の脱落した格助詞と、述部の必須格との関係を調べたものが表2である。横の欄「残り必須格」の数字は述部の必須格の数から、省略されていない文節の修飾によって埋まった必須格の数を引いたものである。脱落した助詞が必須格である場合とそうでない場合を区別してある。また、括弧の中は単語の意味の制約によって文脈によらずに省略された格を推定できる場合の数である。この表から、脱落するのは必須格がほとんどであり(93%)、その約半数が、1格1文節の制限によって、その格を推定できることを示している（残り必須格が1の場合）。さらに、残りの半数は単語の意味と選択制限を使うことによって推定できる。任意格が省略されることは少ないが、省略された格を推定するのは意味を使っても困難であり、文脈処理などのより高度な処理が要求される。

表2 係り先の必須格と脱落した格の関係

残り必須格	0	1	2	3	4
必須格	0 ----	50 ----	53 (48)	11 (8)	0 (0)
任意格	1 (1)	6 (3)	3 (0)	0 (0)	0 (0)

2.3 倒置^{[1][2]}

倒置を分析するために使用したデータは、一字一句、正確に音声と対応がとれていなくても大きな問題はないため、前節までで使ったデータに加えて、テープ起こしの精度があまりチェックされていないものも含めて行った。1文内で、後ろから前に係っている文節を倒置であるとした。表3に倒置された文節の種類（係り方による分類）といくつ前の文節に係っているかをまとめたものを示す。

表3から言えることは、

(1)述部に係る文節がほとんどである（「の」以外のすべて）(95%)。

(2) 1つ前の文節に係る文節がほとんどである(86%)。3つ以上前の文節に係る倒置はない。また、2つ前に係る6つの文節も、前にある2つの文節を慣用的に1つの文節として考えることができるもの(3つ)、2つの文節が倒置されているために一方が2つ前の文節に係っていることになった文節(1つ)がほとんどである。

分析時の問題点は倒置の認定は文の認定基準からの影響を大きく受けることである。倒置された文(節)の前で文が一旦切れて、倒置された文節に続く文が省略されたものと考えれば、倒置でなくなる。

表3 倒置の分類

倒置された文字列の種類	直前	2つ前	合計
が格で係る文節	3	0	3
を格で係る文節	1	0	1
に格で係る文節	0	0	0
で格で係る文節	7	1	8
と格で係る文節	1	0	1
まで、から格で係る文節	1	1	2
は(主題)で係る文節	2	0	2
「～は～です」の倒置	3	0	3
副詞文節	6	0	6
条件文	*1) 6	2	8
「の」で係る文節	1	1	2
合計	31	5	36

直前: 1つ前の文節に係っている倒置

2つ前: 2つ前の文節に係っている倒置

*1) 係り先が省略されている2つの倒置を含む。

表3から明かなように、倒置文節の係り先は容易に決定することができる。問題は、ある文節が倒置であるかどうかを判断する方法である。倒置であると判断された全ての文で、倒置文節は文の最後に存在していた。これは、文の認定基準に大きく依存するが、字面から見る限り妥当な文の区切りとなっている。また、倒置された文節の前で文が一端終了しているという特徴があるため、倒置された文節か否かの判定は容易に可能であると思われる。よって、最後の文節の前に文が一旦終了しており、その後に文を終端させない文節があったとき、その文節は1つあるいは2つ前の述部に係ることができると思えばよい。

2.4 質問－応答対のネストの分析^[3]

我々は音声対話システムを単なる質問応答ではなく、より自然な対話を行えるようにするための一段階として、情報を与える側（多くの場合はシステム）からの質問を許した対話を考えている。この場合、対話に質問－応答がネストした部分が生じる。ユーザが完全で曖昧でない質問をすれば、システムからの質問は必要ないが、そのような条件を満たすのはユーザにとって大きな負担となる。また、現状では計算機の曖昧性を解決する能力が人間よりも悪いため、システムが解決できない曖昧性は積極的に人間に質問するようなシステムを考えねばならない⁽¹⁾。

この節では、音声対話とキーボード対話に対して質問－応答がネストした部分についての分析を行う。分析の対象としたネストした部分対話は、質問とそれに対応する応答の対が入れ子になっているものの中で、外側の質問－応答対の話者が入れ替わっているものである。例えば、質問に答える側が質問した場合などである。質問－応答対が入れ子になっているパターンを図1に示す。あいづちなどの付加的な発話は無視した。上記の定義により、今回対象としたパターンはtype1とtype2である。図中の応答から質問への矢印はその質問－応答の対応を表す。また、[]で囲まれている発話は省略可能なパターンであることを示す。

図1中の点線で囲まれた部分を1つとして数えた場合、2.1節で述べた対話資料の中にネストした対話は87箇所存在した。ネストの分類を情報を与える側からの質問の目的で分類すると次のようになる。

(1) 詳細化の質問

(1-1) 質問に答えるために必要な情報を得るための質問

(1-2) 質問に対する答えのレベルを決めるための質問

(1-3) 質問の意味または対象が曖昧な場合の質問

(1-4) 応答に対するより詳細な情報を得るための質問

(2) 単語の意味に対する質問

(3) 音声聞き取れなかったために生じる質問

(4) その他

それぞれに分類された発話の数を表4に示す。

情報を与える側からの質問は対話全体の発話文の数から比べると非常に少ない。しかし、我々の対話資料では44対話中31対話でネストが1回以上出現しているため、無視できない対話の性質といえることができる。また、キーボード対話では特に少ないのは、文をタイプするのに時間がかかる分考える時間があるのでユーザが比較的完全な質問をしているからだと思われる。

また、今回は意味的な関係は無視したが、意味的なネスト（談話構造）を考えると各質問－応答対は何らかの関係を持っているため、ほとんど全ての対がネスト構造を持っていると言ってよい。しかし、単純な対話制御部を作成するには、ここで扱ったような表層レベルに近いネストを考慮すれば十分であると思われる。

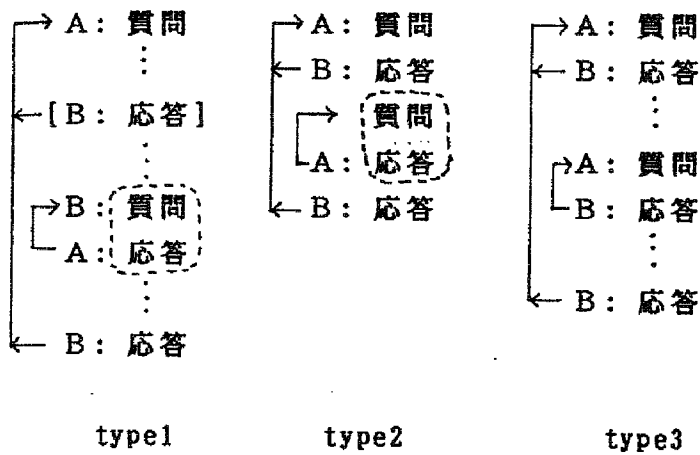


図1 ネストのパターン

表4 ネストの分類

分類	音声		キーボード	
	type1	type2	type1	type2
1-1	50	6	2	0
1-2	1	5	0	0
1-3	9	0	3	0
1-4	6	0	0	0
2	1	0	0	0
3	2	0	-	-
4	2	0	0	0

ネストしている対話の約90%が質問に答えるための詳細化の質問であることが分かった。ユーザの曖昧な質問に対して、その曖昧さを解消するための質問を生成するメカニズムを3.3節で述べる。

音声聞き取れないための確認の質問は当初、多いであろうと予想していたが、ほとんど生じていない。しかし、システムが音声認識装置に基づくとすれば、無視できない重要な問題となる。

次章では、我々が開発してきたキーボードによる対話システムについて述べる。

3. 対話システム

我々は音声による自然言語対話システムを構築中である。アプリケーションとしては、ユーザがある情報を観光案内のデータベースから検索するシステムを考えている。音声認識システムと自然言語対話システム（キーボード入力）の開発を平行して進めてきた。ここでは、自然言語対話システムについて述べる。

3.1 対話システムの全体構成^[6]

図2に対話システムの処理の流れ図を示す。対話システムへの入力はず、日本語解析部で構文・意味解析される。日本語解析部、日本語辞書、文法辞書は以前我々が開発した質問応答システム^[7]を拡張したものを使用している。解析部は意味ネットワークを出力する。次に、入力文の種類の識別を行ない、疑問文であればそれに対する応答文を生成する。生成できない時は対話ルールインタプリタを起動して、対話文を生成する。

入力が平叙文であるとき、この入力がシステムの質問に対する応答であるときには応答から必要な情報を取り出して文脈情報を蓄積する。そして、対話インタプリタを起動して対話文を生成する。入力がシステムに対する応答でないときも対話ルールインタプリタを起動して対話文生成を行う。

次に本システムの特徴の1つである対話ルールとそのインタプリタについて

述べる。

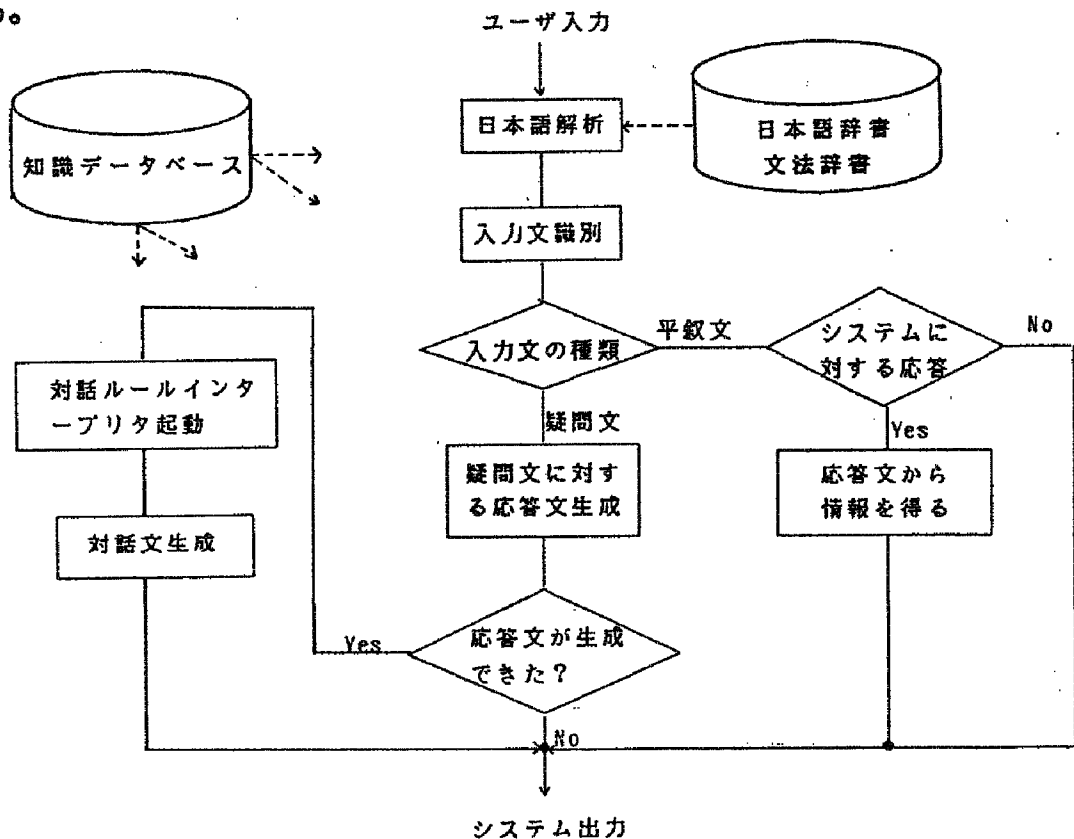


図 2 対話システムの処理流れ図

3.2 対話インタープリタ^[6]

対話ルールは図3の構文にしたがって図4のように記述される。この対話ルールは状況に応じてアドホックに書かれている。

この対話ルールは対話が始まったときに、必ず1つは選ばれていなければならない。対話システムは、ユーザの入力から対話の目的を認識して、目的にあった対話ルールを選択する。そして、対話ルール用スタックに蓄える。例えば、この対話システムは観光案内を対象としているので、「～へ行く」、「～について知る」というような目的がほとんどである。「～へ行く」のような目的のときには図4に示されたいような対話ルール「R_行く」が選択される。

対話ルールが選択されたとき、対話ルールのスロットは必ず埋められなければならない。スロットに埋められる値は入力文の解析結果の意味ネットワークからスロットと同じ属性情報を取り出した値である。

図4の対話ルールの例「R_行く」はスロット値val1からval2へ行くことに関する対話のための対話ルールで、スロット値val1とval2には実際の地名が代入される。例えば、「豊橋から富士山まで行きたい。」のような文が入力されたときは、対話ルール「R_行く」が選択され、スロットfromの値val1には「豊橋」が、toの値val2には「富士山」が代入される。

この対話ルールの第1番目のルール「(if (asked 交通手段) (tell 交通手段))」は、交通手段について聞かれているとき、知識データベースから交通手段の情報を取り出し、ユーザに対して「豊橋から富士山」への交通手段を教える。第2のルールは、ルートについて聞かれているとき、システムがユーザの交通手段を知っているときルートを教え、知らないときには「R_ask交通手段」という別の対話

対話ルールの構文:

```
対話ルール: (対話ルール名 (スロット . . . )
              命令
              .
              .
              )
スロット: (スロット名 スロット値)
```

対話ルールの命令:

(if (条件文) (実行1) (実行2)): 条件文が真のとき実行1、偽のとき実行2を実行する。
(tell A): ユーザにAに関することを教える応答文を生成する。
(ask wh A): 疑問代名詞 wh を用いてAに関する疑問文を生成する。
(call 対話ルール名): 新しい対話ルールを起動する。

対話ルールの条件文

(asked A): Aについて質問されているとき真となる関数。
(know A): Aに関する知識をユーザがすでに知っている場合に真となる関数。

図3 対話ルールの構文と命令

```
(R_行く ((from val1) (to val2))
  (if (asked 交通手段) (tell 交通手段))
  (if (asked ルート)
    (if (know 交通手段) (tell ルート)
      (call R_ask交通手段 val1 val2)))
  .
  .
)
```

図4 対話ルールの例

ルールを呼び出し、交通手段を尋ねる。

対話ルールインタープリタに入力があると、対話ルールインタープリタが起動して対話ルールを解釈・実行する。対話ルールの実行により対話文が生成できれば、そのまま文を出力してユーザの入力待ちになる。対話文が生成できなかったときは、対話ルールが失敗したので対話ルールスタックの一番上の対話ルールを取り去る。そして、「分かりません」と出力する。

次の節では、この対話システムをより自然な対話ができるように拡張する1つの方法について検討と試作について述べる。

3.3 入力文の曖昧さを解消するための質問生成システム^[8]

2.3節で述べたように、計算機による自然言語対話システムを考えた場合、現状では曖昧性を解決する能力が人間よりも悪いため、システムが解決できない曖昧性は積極的に人間に質問するようなシステムを考えねばならない^{[7][9]}。

本節で述べるシステムは入力として意味表現を受け取り、文脈などの知識を用いて省略を補完し、文脈に入力の意味表現の要素を対応させた拡張された意味表現を出力する。この拡張で解消できない曖昧性が生じた場合にその曖昧性を解消するための質問を生成する。また、曖昧性がなくても、質問に答えるための情報が欠落している場合は、その情報を得るための質問を生成する。

次の対話は「どのくらいかかりますか?」が曖昧な解釈を持つ例である。

A 1: 豊橋から大月に行きたいのですが、大月はどこにありますか?

B 1: 山梨県の東にある町です。

A 2: どのくらいかかりますか?

B 2: 時間がですか? お金ですか?

この例では、A2の質問の対象が時間であるのかお金であるのかが曖昧であるため、その曖昧さを解消するためのシステムの質問が生成されている。もちろん、A2の発話の前に時間の話をしていた場合は、文脈的な処理によって時間ということを決断する。処理の順序を以下に示す。

- (1) 文脈への埋め込み
- (2) (wh型疑問文の場合は) 質問の対象の決定
- (3) 答えるのに十分な情報があるかどうかのチェック

各処理段階で曖昧性が生じたら直ちに質問が生成される。ここでは、紙面の都合上、(2)の「質問の対象の決定」のみについて説明する。

3.3.1 使用する知識

曖昧性を解消するために用いられる知識は、(1)文脈、(2)動詞スロット辞書、(3)疑問代名詞スロット辞書である。以下に簡単に説明する。

(1) 文脈

処理している発話が発せられる前までの対話によって対話参加者が共通に持っているであろうと考えられる知識を表す。すなわち、共有知識としての文脈である。ここで考えている文脈は話の内容のまとめではなく、省略の補完に主に使われるものであり、各情報の関連を表しているに過ぎない。疑問文は平常文の形で文脈に残される。

(2) 動詞スロット辞書

各行為の概念が取ることが可能な格スロット名のリストから成る。例えば、「行く」の動詞スロット辞書は以下のようになる。

(GO agent from to instrument cost time)

これは、GOという概念がagent(主格), from(始点格)などを取ることができることを表している。

(3) 疑問代名詞スロット辞書

疑問代名詞の種類によって質問の対象が制約される。例えば、「どのくらい」という句は疑問の対象として何か量に関係あるもの(例えば、値段や時間など)でなければならないという制約となる。このような疑問代名詞の種類はパーザによって意味表現の一部として表現される。値としてはHOW-DEGREE(どのくらい)、HOW-METHOD(どのように)などがある。制約がないものはWHATという値をとる。

この知識は、疑問代名詞の種類と、その疑問代名詞が質問の対象としてとることができる格スロット名のリストにより表現される。例えば、以下のようになる。

(HOW-DEGREE height mass cost time) --- (「どれくらい」など)

(HOW-METHOD way instrument route) ---- (「どのように」など)

(WHO agent patient recipient) ----- (「だれ」など)

3.3.2 質問の対象の決定

wh型質問の対象を決定するために、文脈、動詞スロット辞書、および疑問代名詞スロット辞書を使用する。処理はまず次のような対象の集合を求める。

(1) 疑問代名詞の種類から決定される対象の候補の集合(疑問代名詞辞書を使用)

(2) 疑問文の中心的な動詞から許される対象の集合(動詞スロット辞書を使用)

この2つの集合の積集合を質問の対象の候補とする。次に、文脈への埋め込みを試みる。対象の候補があるので、その対象に関連して文脈へ最も適合する対象を評価する。さらに、文脈を参照して対象の値がすでに知られているスロット名は候補から削除する。

結果として最も確からしい候補が1つになった場合、それがただ1つの候補として次の処理へ進む。また、1つ以上になった場合はそのうちのどれが質問の対象かを尋ねる質問を生成する。

4. おわりに

音声対話特有の現象である付属語の脱落、倒置、詳細化の質問を分析した。また、我々の研究室で開発してきたタイプ入力の対話システムについて述べ、そのシステムが曖昧な質問に対して曖昧さを解消するための質問を生成するための拡張についても述べた。

今後の課題は、音声認識システムとタイプ入力対話システムの結合を行うことである。どちらのシステムも実時間性を考慮しており、ユーザの入力から20～30秒で応答することを目標としている。このシステムの開発と実験により実用化に対する新しい知見が得られると思われる。

謝辞

日本音響学会連続音声データベースを使用させていただきました。作成された方々にお礼を申し上げます。

参考文献

- [1] 小林, 山本, 中川, 「音声対話におけるいくつかの統語的特徴」,
1991年度電気関係学会東海支部連合大会, pp.556, 1991.
- [2] 山本, 小林, 中川, 「音声対話文における助詞落ち・倒置の分析とその解析手法」
準備中.
- [3] 山本, 小林, 中川, 「対話における質問応答対の入れ子構造の分析」,
1991年電子情報通信学会秋季大会, D-54, 1991.
- [4] 有田, 小暮, 野垣内, 飯田, 「メディアに依存する会話の様式——電話会話とキーボード会話の比較——」, 情報処理学会研究報告, 87-NL-61-5, 1987.
- [5] 井ノ上, 江原, 小倉, 「係り受け関係データから見たキーボード会話と電話会話の比較」, 情報処理学会第40回(平成2年度前期)全国大会, 6F-3, pp.490-491, 1990.
- [6] 田口勝豊, 「日本語による質問・対話システムに関する研究」,
豊橋技術科学大学修士論文、1991年2月.
- [7] 中川, 竹本, 田口: 「交通規則文に関する質問応答システムLICENCEにおける日本語文から一階述語論理式への変換」, 情報処理学会論文誌, 第32巻, 第3号, pp.354-363, 1991.
- [8] 山本, 小林, 中川: 対話における曖昧さを解消するための質問生成システム」,
人工知能学会全国大会、13-8, pp.551-554, 1991.
- [9] Niedermaier, G.Th., Streit, M. and Tropic, H. :
Linguistic processing related to speech understanding in SPICOS II,
in Speech Comm., 9, pp.565-585, 1990.

音声対話における不要語・復唱・倒置 および助詞の役割について

小林 哲則 白井 克彦

(早稲田大学・理工学部)

1. まえがき

従来、音声処理では、一般に不要語（無意味語）や倒置は扱うことが難しいとの理由で無視されており、また、助詞は無視すると探索空間を削減する有用な構文上の制約を設けることが難しいとの理由で厳格に扱われている。しかし、一般に不要語とされている語が伝える情報は無視できないものがあり、また、口語では言い誤りや省略がつきものともいえる助詞を頼った処理は本質的に限界を持つものと筆者等は考えている。本報では、音響学会連続音声データベースの模擬対話の書き起こしテキストに見られる例を眺めながら、

- 1) 不要語、間、韻律、復唱が伝えるものはなにか？
- 2) 倒置は主部、述部が入れ替わっただけか？
- 3) 助詞はどれだけの情報量を持っているのか？

といった問題について考えてみる。

2. 不要語・間・韻律・復唱の役割

2.1 不要語・間・韻律

不要語と呼ばれる語をその機能から次のように分類できると考えられる。

1) 対話の番に関するもの

1-1 話しの番をとるためのもの（語りかけ）

「あの一」

意味：これから話しかけますよ！

1-2 相手が安心して話しの番を保持できるようにするためのもの（あいづち）

「ええ」「はい」（「うっそお」「ほんとお」）

意味：聞いていますよ！

極弱い、合意、了解、時として軽い意外感の表明。

1-3 相手に番を渡さないためのもの（自己調整）

「えーと」

意味：まだ話しが続くんですけど、考えてるんでちょっと待って下さい。

話しにつまってしまったので、ちょっと言い直します。

2)積極的に意味を伝えるもの（特に韻律と組みで、時として間として。）

2-1 意外感を伝えるもの

「えっ」

2-2 困惑感を伝えるもの

「んー」

これらの中でも、1-3の不要語は口語文の解析上重要な意味を持つと考える。次の例文を見てみよう。

<対話例 1>

Q：だからもし、安く、[えーと] ご予算にもよるんでね、どれくらいで。

この「えーと」は、「少し待って下さい。言い直します。」くらいの意味である。聞き手はこの語を鍵として、前の文が行き詰まって言い直されたことを意識することが可能になり、前の部分と後ろの部分を切り放して文を理解することができる。この語なしに、すらすらと上の文全体を話されたなら、人間が聞いたとしても瞬時の理解は困難であろう。

他の例を見てみよう。

<対話例 2>

Q：[あの一] バイクの免許を取りたくて、

[えーと] ちょっとお聞きしたいんですけど。

A：ええ。

Q：どのような方法、

[えーと] ちなみに私普通免許の免許を今持っているんですけど。

A：で、どこで取ります。

Q：[えー] 特に教習所は指定しませんが、

教習所以外にもいい方法があれば教えていただきたいんですけど。

この例の場合は、上から順に「今から話しかけます」、「ちょっと待って下さい。まだ話しますよ。」、「ごめんなさい。言い直します。」、「そうだな、ちょっと考えさせて下さい」くらいの意味を伝えている。

<対話例 3>

Q：この仕事やっといてくれないかな。

A：[うーん] いいですよ。

<対話例 4>

Q：新宿の名産品といったらなんですか。

A：[うーん] 新宿のっていわれると少し困ります。

これらの例は、それぞれ、「こまったなあ。本当はいやなんだけどなあ。」、「こまったなあ。新宿に名産品なんてありゃしないし・・・」くらいの意味であろう。上の分類でいう2-2に属す不要語である。人間と機械との間での高度なコミュニケーションを目指そうとするならば、これらの語の発話（声）に現れる話者の心情の扱いは不可避である。これらの発話においては、間・韻律も大きな情報を持っており、発話媒介というよりは、発声そのものが媒介して意味・心情を伝える行為といえよう。

こうして、その不要語と呼ばれる語の機能を考えてみるといづれも簡単に無視しうるものではないことが理解できよう。即ち、不要語の抽出（解析）が必要なのは、意味理解するにあたって邪魔だから取り除くためなのではなく、より正確な意味理解のため必要だからである。

2.2 復唱の役割

対話データを眺めみると、復唱の多いことに気がつく。そして、この復唱もまた、不要語と同様に多くの情報を伝えている。

例を見てみよう。

<対話例 5>

Q：機械振興会館というらしいんですが、場所全然知らないんで。

A：機械振興会館は、簡単にいっちゃうと東京タワーの前です。

Q：東京タワーの前。

この復唱には、「急にそんなこと言われたって東京タワーなんてどこだか知らないよ」くらいの意味が感じられる。

<対話例 6>

Q：吉祥寺でお食事をしたいんですけど、いいお店教えてください。

A：いいお店ですか。

Q：あんまり高くなくて、で、静かで、雰囲気の良いお店で、美味しい。

A：具体的にいうと、どんなものを食べたいんですか。

Q：[うーんとね] 洋食。

A：洋食。

Q：うん。

A : [んー] 静かで、雰囲気の良い（洋食の店）。

まず、吉祥寺にあまり静かな店無いんです。

Q : ない。

これらの例は、それぞれ、「もっと具体的に言ってくれなきゃ答えようがないよ」、「ちょっとまってくださいよ。いま考えますから・・・。困ったなあ。案がないなあ。」、「ええっ！ ないのぉ!!」くらいの意味を伝えている。

要するに、復唱は確認とともに、意外感の表明、不服の表明、などの役割を演じている。

3. 倒置あるいは断片的な情報提示について

つぎの例を見てみよう。

<対話例 7>

Q : パスポートを取るのにどのくらいかかりますか？

土浦のほうに行きたいんです。

即日じゃないという話は聞いたんですけど、

これは、次のような文が倒置されていると考えるべきではない。

<対話例 8>

Q : パスポートを取るのにどのくらいかかりますか？

即日じゃないという話は聞いたんですけど、

土浦の方に行きたいんです。

この解釈では、即日じゃないことを知っているにも関わらず、土浦の方へ行きたいと言っているように受け取れる。この例では、

<対話例 9>

Q : 土浦の方にパスポートをとりに行きたいんです。

この場合即日じゃとれないという話は聞いたんですけど、

どのくらいかかりますか？

という内容を、

<対話例 10>

Q : パスポートを取るのにどのくらいかかるか知りたいんですけど。

そうそうパスポートがとれるところはいくつかあるんですね。
 とりあえず、私が行きたいのは土浦の方なので、この場合を教
 えて下さい。

この場合、即日じゃないとれないことまでは知っていますよ。

という順序で提示したものと考えるのが自然である。このように、会話では、断片的な情報の提示によって、従たる情報の付加を行い、日本語におけるトップダウンな情報伝達を実現している。倒置も基本的には、この情報の断片的提示と考えるべきであり、そこには、情報の主従関係の提示が含まれている。倒置に目を向けないことは、これらの主従関係を直接発話から得る機会を失っていることに等しい。

4. 助詞の情報量について

助詞は音声認識において探索空間の削減に有用な構文制約を設けるうえで重要な鍵を握っており、この脱落や言い誤りを認めずにシステムを構築することが多い。しかし、一般に口語では助詞の言い誤りや省略はつきものであり、これに大きく依存した処理系を構築することには疑問を持たざるを得ない。そこで、ここでは、そもそも助詞は対話においてどの程度の役割を演じているかを調べるため、次のような実験を試みた。

実験では、3つの対話A、B、Cの書き起こしテキストから助詞を除いたテキストを10人の被験者に提示し、助詞を補うことを求めた。助詞を決める際には、その箇所以前の文章は参照してもよい（即ち、その状況に至るまでの対話の流れは参考にしてよい）ものとした。結果として得られた回答のバリエーションを表1に示す。

表1 助詞の穴埋めテストにおける回答のバリエーション

回答の個数	頻度（同意の助詞も区別）			頻度（同意の助詞を区別せず）		
	A	B	C	A	B	C
1	21	27	31	58	60	76
2	25	20	25	3	0	0
3	11	9	14	0	0	0
4	1	4	2	0	0	0
5	2	0	4	0	0	0
6	1	0	0	0	0	0

(回答の個数とは、回答のバリエーションの個数を意味する。)

この結果から分かるように、助詞が一意に決まらなかった例は3カ所のみであり、対話の状況が明確につかめるとき、助詞のエントロピーは零に近いことが分かる。即ち、各単語の役割は、単語間の可能な意味的關係と状況だけでかなりの部分が決定できる。一般の言語処理では、与えられた助詞をもとにそれがついた単語の文内における意味的役割を決め、最終的に求まった意味から状況を推測するが、その逆のプロセスも重要であり、両者がほどよくかみあって始めて有効な言語処理システムは実現できるものと考えられる。

5. むすび

不要語、復唱が伝えるものはなにか、倒置は単純に主部、述部が入れ替わっただけと考えてよいか、助詞はどれだけの情報量があるか、等について検討し、不要語、倒置を積極的に発話内容の理解に役立てていくことの重要性と、助詞に依存しないで処理系を構築することの可能性・重要性を指摘した。ここで行った対話の分析はまだまだ一面的なものであることは否定できず、議論の余地を多く残すものではあるが、今後会話音声の理解システムをより実質的なものへとレベルアップするために必要とされる要因を議論する種になるものとする。

謝辞

本研究には、音響学会連続音声データベースの模擬対話書き起こしテキストを使用した。

日本語文音声認識のための統語処理アルゴリズム —機能語予測CYK法—

伊藤 彰則 牧野 正三
(東北大学 応用情報学研究センター)

1 はじめに

音声による機械との対話や音声による文章入力などを実現するためには、音声の持つ言語的な性質を最大限に利用してその音声の認識・理解を行う必要がある。日本語文章音声を対象とした統語処理アルゴリズムとしては、文節認識と文節間の係り受け解析に基づく方法[3][4]と、文脈自由文法を用いる方法[1][2][6][10]に大別される。日本語には「文節内文法と文節間文法が比較的きれいに分離できる」「先行要素が後続要素を修飾する」などの特徴があるため、これらを考慮すれば、文法の記述の容易さなどの点で文節内・文節間の2段階処理の方が有利である。しかし、文節認識と係り受け解析に基づく方法は、その多くが上昇型の解析法であるために、文法的な制約を利用して単語候補を絞り込むということができない。そのため、認識精度・計算量の点で不利になる。

そこで、文法的には文節内文法と文節間文法を用いながら、全体としてtop-downな処理を導入すれば、文法の記述の容易さと処理の効率化を同時に実現することができる。本研究では、このような特徴を持った統語処理アルゴリズム「機能語予測CYK法」を開発した。本稿では、この機能語予測CYK法の概略と、認識実験による有効性の検討について述べる。

2 機能語予測CYK法

ここでは、機能語予測CYK法の考え方と基本的なアルゴリズムについて述べる。機能語予測CYK法では、基本的なCYKのアルゴリズムに加えて、機能語の予測を行う。まず初めに、文の解析に用いる文法を定義する。

2.1 解析に用いる文法

文の解析に用いる日本語の文脈自由文法を $G = (N, T, P, S)$ とする。ただし、 N, T, P, S は、それぞれ非終端記号の集合、終端記号の集合、書換規則の集合、始端記号である。

終端記号の集合 T を、実質語 T_c と機能語 T_f に分割する。 $T_c \cup T_f = T$ かつ $T_c \cap T_f = \emptyset$ である。 T_c は名詞、動詞（語幹）、形容詞（語幹）、形容動詞（語幹）、副詞、連体詞の集合であり、 T_f はそれ以外の単語（あるいは語尾）の集合である。（「また」「しかし」などの接続詞は、副詞に含めて考える。）このとき、書換規則 P は、次の形の規則のみを含むとする。

- (Type A) $\alpha \rightarrow \beta\gamma$ ($\alpha, \beta, \gamma \in N$)
 (Type B) $\alpha \rightarrow \beta a$ ($\alpha, \beta \in N, a \in T_f$)
 (Type C) $\alpha \rightarrow a$ ($\alpha \in N, a \in T_c$)

Type A の規則は文節間文法、Type B の規則は文節内文法、Type C の規則は語彙規則にそれぞれ対応する。

2.2 機能語予測 CYK 法

まず、機能語予測 CYK 法の基本的な考え方について述べる。ここで解析に用いる文法 G は、Type A および Type C の規則のみに着目すると、Chomsky 標準形になっていることがわかる。そこで、Type A と Type C の規則については CYK 法を用いて解析し、Type B の規則は別に解析を行う。Type B の規則の右辺の 2 個目の要素は必ず終端記号であるから、Type B の規則については直接的な単語の予測が可能である。この性質を利用し、Type B の規則を処理する場合には、その後にくる可能性のある機能語を予測する。

ここからもわかるように、機能語予測 CYK 法では、機能語についてしか予測を行わず、実質語については予測動作を行わない。これは Earley 法や LR 法に比べて不利に思える。しかし、対象を一般的な日本語に限った場合、少なくとも文法的には、実質語の予測は困難である。そのため、Earley 法や LR 法によって単語の予測をしたとしても、実際に予測が可能なのは機能語に限られることになる。したがって、単語の予測機能を持つ手法と機能語予測 CYK 法を比較しても、予測効率に大きな差はないものと考えられる。

2.3 ビームサーチの導入

通常書き換え規則を用いて文法の記述を行うシステムでは、この「CYK 法に機能語の予測を加える」という基本的なアイデアをそのまま用いることができる。しかし、近年の音声認識においては、言語的な制約を認識精度の向上につなげるため、統語的・意味的・談話的な制約を低レベルの探索過程に反映させる方法が主流となってきた。このような処理を効率的に行うため、JPSG[7]に代表されるような単一化文法に基づく文法記述法が多く用いられている。このような文法

においては、従来の文脈自由文法の非終端記号にあたるものが素性の束として表され、等価的な非終端記号数が非常に多くなる。そのため、従来の統語解析法である Earley 法、チャート法、拡張 LR 法などをそのまま用いると、解析のための記憶容量や計算量が膨大なものとなる。そこで、機能語予測 CYK 法にビームサーチを導入することにより、この問題を解決する。ビームサーチを導入することによって解の最適性は失われるが、記憶領域と計算量を大幅に削減することができる。また、Type B の規則 $(\alpha \rightarrow \beta a, \alpha, \beta \in N, a \in T_f)$ については Earley 法などと同じく規則の選択の必要があるが、機能語オートマトンを導入することによってこの問題を回避する。

2.4 機能語オートマトンの導入

2.2 で述べたアルゴリズムでは、Type B の規則 $\alpha \rightarrow \beta a$ によって機能語の予測を行う。しかし、規則の数が多くなると、ある β についてすべての $\alpha \rightarrow \beta a$ をチェックするということが難しくなってくる。そこで、次のような性質を持つ比較的小規模な正規文法 $G_f = (N_f, T_f, P_f, S_f)$ を使って機能語の予測を行う。(これは有限オートマトンと等価である。この正規文法を有限オートマトンとみなした場合、オートマトンの状態遷移図におけるノードは正規文法における非終端記号に対応し、またアークは正規文法の書換規則に対応する。以下、この正規文法を有限オートマトンとみなし、機能語オートマトンと呼ぶ。)

(性質 1) すべての Type C の規則 $\alpha \rightarrow a \in P$ について、 $\psi(\alpha) = \text{cat}(a), S_f \rightarrow \psi(\alpha) \in P_f$ である。ただし、 $\text{cat}(a)$ は $a \in T_c$ の品詞を表し、 $\text{cat}(a) \in N_f$ である。また、 ψ は N から N_f への写像であり、 G の非終端記号と G_f の非終端記号との対応を表す。

(性質 2) すべての Type B の規則 $\alpha \rightarrow \beta a \in P$ について、 $\psi(\alpha) \in N_f$ である。また、 α を右辺に持つ Type B の規則が P に含まれていれば、 $\psi(\beta) \in N_f, \psi(\beta) \rightarrow a\psi(\alpha) \in P_f$ 、そうでなければ $\psi(\beta) \rightarrow a \in P_f$ である。

(性質 3) すべての Type A の規則 $\alpha \rightarrow \beta\gamma \in P$ について、 $\psi(\alpha) = \psi(\gamma)$ である。

これらの性質は、この正規文法によって行われる機能語の部分的な解析が、もとの文法による機能語の解析と一致することを表している。

2.5 アルゴリズム

ビームサーチを導入した機能語予測 CYK 法のアルゴリズムを示す。入力としては、連続発声された文音声から音素認識を行って得られた音素系列、あるいはフ

レーム毎に入力ボタンと音素標準ボタンをマッチングした結果などを想定している。マッチング結果は尤度で表され、これを最大化する。

まず初めに、アルゴリズム記述のための記号を定義する。解析のための文法を $G = (N, T, P, S)$ とし、Type A, B, C の書き換え規則の集合をそれぞれ P_A, P_B, P_C とする。また、Type A, B, C の規則の左辺に現れる非終端記号の集合をそれぞれ N_A, N_B, N_C とする。 ε は空列、 ω は N に含まれない記号とする。（ ω は非終端記号に対応しない部分木を表すために使用する。）文法 G に属する非終端記号 α と単語列 w 、機能語オートマトンのノード（非終端記号） α_f および尤度 l の組

$$\langle \alpha, \alpha_f, w, l \rangle, \alpha \in N \cup \{\omega\}, \alpha_f \in N_f \cup \{\omega\}, w \in T^*$$

を部分木と呼ぶ。また、部分木 $t_1 = \langle \alpha_1, \beta_1, w_1, l_1 \rangle$ と $t_2 = \langle \alpha_2, \beta_2, w_2, l_2 \rangle$ を規則 $\gamma \rightarrow \alpha_1 \alpha_2$ で融合した部分木を $U(t_1, t_2) \equiv \langle \gamma, \beta_2, w_1 w_2, l_1 + l_2 \rangle$ と定義する。その他に、次の記号を定義する。

I	入力系列長
D	部分木のビーム幅
$X_{i,j,d}$	始端 i 、終端 j で、 d 番目の尤度の部分木 ($1 \leq i \leq j \leq I, 1 \leq d \leq D$)
$Sc(j, i w)$	入力系列 j から i までと終端記号 w とのマッチング尤度
θ	マッチングの閾値
$PC(i)$	位置 i で終端する部分木の非終端記号の集合
PR	ある位置で予測される Type B の規則の集合

また、次の動作を $puttree(i, j, t)$ と略記する。

入力系列の i から j の位置にある部分木の集合 $S_{i,j} = \{X_{i,j,d} | 1 \leq d \leq D\}$ に部分木 t を加えた集合について、その $D+1$ 個の要素を尤度順にソートし、上位 D 個を新たに $X_{i,j,1} \sim X_{i,j,D}$ とする。ただし、 $t = \langle \alpha, \beta, w, l \rangle$ として、 $S_{i,j}$ に $t' = \langle \alpha, \beta, w, l' \rangle$ なる部分木が含まれていた場合は、尤度の大きい方を残し、小さい方は $S_{i,j}$ から除く。

以上の定義のもとに、機能語予測 CYK 法のアルゴリズムを図 1 に示す。

機能語予測 CYK 法の処理は、大きく (1) 実質語のマッチング、(2) 部分木の融合、(3) 機能語の予測、(4) 機能語のマッチングに分かれる。(1) と (2) の部分は CYK 法での処理とほぼ同じであり、Backward time alignment [1] による実質語のマッチングと、書換規則の適用を行う。(1) では、ある位置 i を終端として、すべての実質語を始端フリーでマッチングする (Type C の規則の適用)。(2) では、 i を終端とするすべての区間 ($j \sim i$) に注目し、その区間を 2 つに分け、前半にある部分木と後


```

for all  $1 \leq i, j \leq I, 1 \leq d \leq D$  do
   $X_{i,j,d} \leftarrow \langle \omega, \omega, \epsilon, -\infty \rangle$ 
for  $i \leftarrow 1$  to  $I$  do begin
  (実質語のマッチング)
  for all  $\alpha \rightarrow w \in P, \alpha \in N_C, w \in T_c$  do begin
    単語 (終端記号)  $w$  を、終端を  $i$  に固定して始端フリーで
    マッチングする。この結果、 $Sc(j, i|w), 1 \leq j \leq i$  が求まる。
    for  $j \leftarrow i$  downto  $1$  do
      if  $Sc(j, i|w) > (i - j + 1)\theta$  then
         $puttree(j, i, \langle \alpha, cat(w), w, Sc(j, i|w) \rangle)$ 
    end
    (部分木の融合)
   $PR \leftarrow \emptyset$ 
  for  $j \leftarrow i - 1$  downto  $1$  do begin
    for  $k \leftarrow j$  to  $i - 1$  do
      for  $d \leftarrow 1$  to  $D$  do
        for  $e \leftarrow 1$  to  $D$  do begin
           $X_{j,k,e} = \langle \alpha_1, \beta_1, w_1, l_1 \rangle, X_{k+1,i,d} = \langle \alpha_2, \beta_2, w_2, l_2 \rangle$  とする
          for all  $\gamma \rightarrow \alpha_1 \alpha_2 \in P$  do
            if  $\lambda(U(X_{j,k,e}, X_{k+1,i,d})) > (i - j + 1)\theta$  then
               $puttree(j, i, U(X_{j,k,e}, X_{k+1,i,d}))$ 
            end
          for  $d \leftarrow 1$  to  $D$  do begin
             $X_{j,i,d} = \langle \alpha, \gamma, w, l \rangle$  について、 $\gamma$  を  $PR$  に加える。
          end
        end
      end
    end
    (機能語の予測とマッチング)
    for all  $\gamma \rightarrow f \alpha \in P_f, \gamma \in PR, \alpha \in N_f \cup \{\epsilon\}$  do begin
      単語  $f$  を、 $i + 1$  を始端として終端フリーでマッチングする。
      その結果、 $Sc(i + 1, m|f), i + 1 \leq m \leq I$  が求まる。
      for  $j \leftarrow 1$  to  $i$  do
        for all  $\langle \beta, \gamma, w, l \rangle \in \{X_{j,i,d} | 1 \leq d \leq D\}$  do
          for  $m \leftarrow i + 1$  to  $I$  do begin
             $X \leftarrow U(\langle \beta, \gamma, w, l \rangle, \langle \omega, \alpha, f, Sc(i + 1, m|f) \rangle)$ 
            if  $\lambda(X) > (m - j + 1)\theta$  then
               $puttree(j, m, X)$ 
            end
          end
        end
      end
    end
  end
end
end

```

以上の解析が終わった時点で、 $X_{1,I,1} = \langle \alpha, \beta, w, l \rangle$ とすると、 w が認識結果、 l が認識スコアになる。

図 1: ビームサーチを導入した機能語予測 CYK 法

半にある部分木の間でType Aの規則を適用する．ここまでの処理が終わると，位置 i で終端するすべての部分木が求まり，位置 i で終端する部分木に対応する機能語オートマトンのノードの集合 $PC(i)$ ができる．(3)では， $PR(i)$ に基づき，機能語オートマトンを用いて機能語の予測を行う．(4)では，(3)で予測された機能語をマッチングするとともに，その規則を適用して新たな部分木を生成する．これらの処理を行う場合，入力系列のすべての位置について，部分木のカテゴリ（対応する非終端記号）に関係なく D 個の部分木を残す($X_{i,j,d}$)．そのため，ある位置に部分木を登録するときに，部分木のソート(*puttree*)が必要である．このアルゴリズムの計算量は，単語の平均長さを L ，平均予測機能語数を V_p とすると， $O((D+V_p)DI^3 + (|N_c|+V_p)IL^2)$ で表される．

3 比較実験

本アルゴリズムと，文節検出＋統語処理の2段階の処理とを比較するため，評価実験を行った．

3.1 実験条件

実験に用いたテキストは音素認識率8530文，実質語辞書は843単語である．機能語オートマトンは，文献[8]のものを利用した．このオートマトンに含まれる機能語は431語，非終端記号数は106，規則数は559である．

3.2 2段階の統語処理法との比較

機能語予測CYK法と，文節検出＋統語処理の2段階による認識との比較を行った．2段階の処理において，文節の検出には構文駆動型連続DP法[5]を用いた．構文駆動型連続DP法ではパラメータとして閾値が与えられる．また，統語処理にはCYK法に基づくアルゴリズム[9]を用いた．このアルゴリズムではパラメータとしてビーム幅が与えられる．一方，機能語予測CYK法では，パラメータとして閾値とビーム幅の両方が与えられる．そこで，この2つのパラメータを変えて実験を行った．まず，ビーム幅を5,10,15に固定し，その条件下で閾値を変えながら認識実験を行い，単語認識率を算出する．この場合の単語認識率は，実質語・機能語のいずれも1単語として算出したものである．閾値を変えることによって単語認識率が変化するから，この単語認識率をパラメータとしてDPマッチングと統語処理の計算量の比較を行った．結果の図中で，単語認識率が最高85～90%程度となっているが，実験に用いたシステムの計算時間・記憶容量の関係上，これ以下まで閾値を下げた実験は困難であった．

はじめに，機能語予測CYK法と構文駆動型連続DP法について，計算量の比較

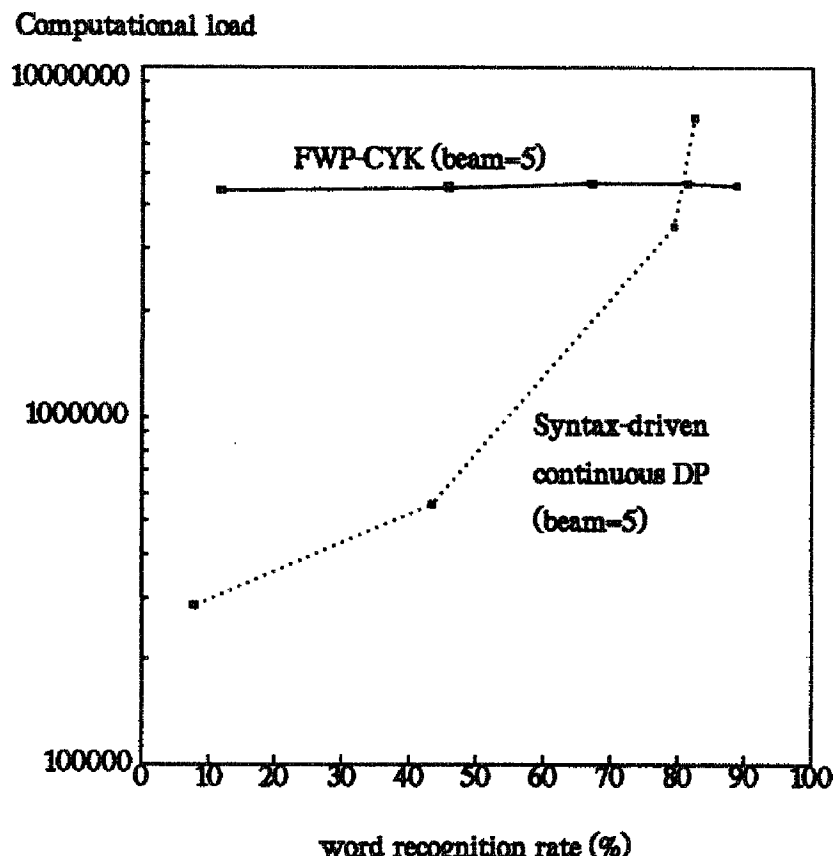


図 2: 単語認識率と DP 計算量

を行った。ここでは、計算量の目安として、DP マッチングの漸化式計算の回数を用いる。それぞれの方法について、単語認識率と 1 文あたりの漸化式計算回数を図 2 に示す。機能語予測 CYK 法（図中の FWP-CYK）では、DP の計算量はビーム幅にはほとんど依存しないことがわかる。また、閾値を変えても計算量はほとんど変わらない。一方、構文駆動型連続 DP 法では、閾値を下げることによって急激に計算量が上昇する。そのため、ある程度以上の認識率を得ようとするとき、機能語予測 CYK 法の方が有利であるといえる。

つぎに、前の実験での文節検出によって生成された文節ラティスから CYK 法を用いて統語処理を行う方法と機能語予測 CYK 法を比較した。ここでは、計算量の目安として、依存関係（係り受け）のチェック回数を用いている。それぞれの方法について、ビーム幅を変えた場合の単語認識率と 1 文あたりの依存関係チェック回数を図 3 に示す。図中の FWP-CYK は機能語予測 CYK 法、CYK は CYK 法による結果をそれぞれ表している。この結果から、機能語予測 CYK 法は CYK 法に比べて計算量が少ないことがわかる。また、機能語予測 CYK 法では、ビーム幅の拡大に伴う計算量の増大が CYK 法に比べて小さく抑えられている。

これらの結果から、ある程度以上の単語認識率を得る場合（この実験では 80% 程度）には、2 段階の処理方式よりも機能語予測 CYK 法の方が有利であると言える。

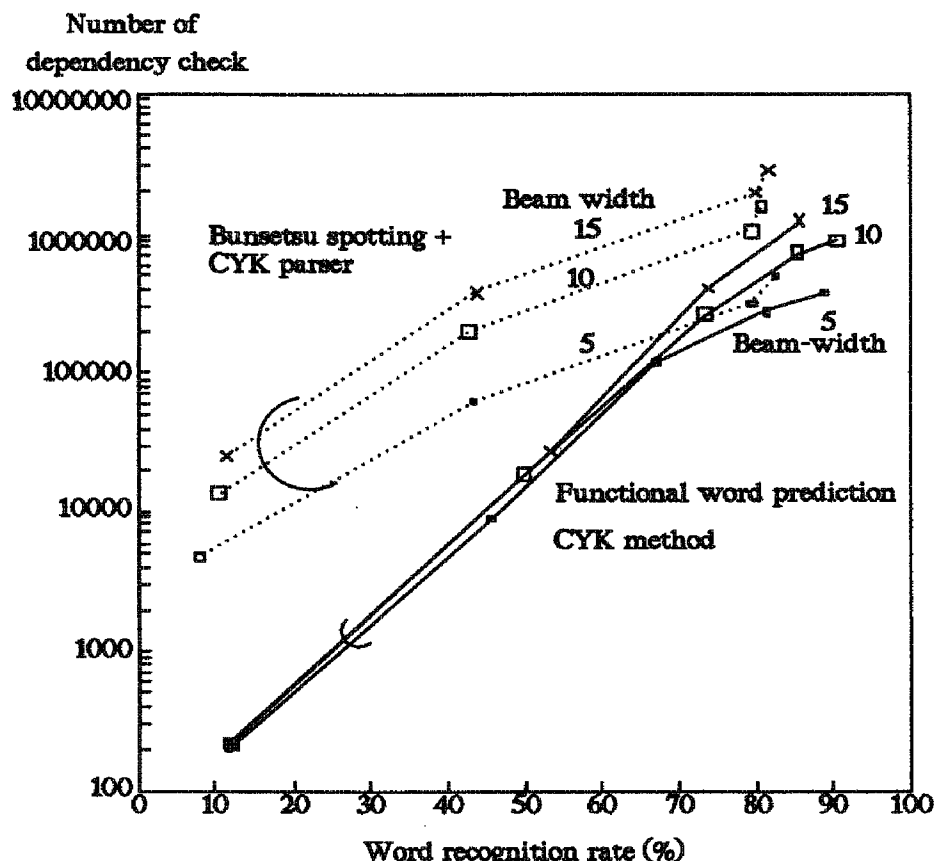


図 3: 単語認識率に対する統語処理の計算量

4 むすび

連続音声認識のための新しい統語処理アルゴリズム「機能語予測CYK法」について述べた。この方法は、従来のCYK法に機能語の予測機能を加えたもので、この予測機能によって効率的な処理を行うことができる。また、非終端記号の増加に伴う記憶容量と計算量の増加を抑えるため、機能語予測CYK法にビームサーチを導入したアルゴリズムを提案した。

機能語予測CYK法と、従来の文節検出+統語処理による認識の比較実験を行った結果、機能語予測CYK法はDPマッチング及び統語処理のいずれの段階においても計算量の点で有利であることが明らかとなった。

今後は、今回比較した従来法以外の方法との比較を行っていくとともに、機能語予測CYK法を実際の連続音声認識システムに組み込んで、その性能の評価を行っていく予定である。

参考文献

- [1] H.Nay: "Dynamic Programming speech recognition using a context-free grammar", Proc. ICASSP87, pp.69-72 (1987)
- [2] 中川: 「文脈自由文法のフレーム同期型構文解析法による連続音声認識」,

信学論 D J70-D,5, pp.908-916 (1987-5)

- [3] 中川, 伊藤: 「音節標準パターンと逆時間向き係り受け解析法を用いた日本語文音声の認識」, 信学論(D),J70-D,12,pp.2469-2478 (1987-12)
- [4] 岡田, 松尾, 伊藤, 小川, 牧野, 城戸: 「文節スポッティングに基づく日本語文音声の認識」電学論 108-C,10, pp.826-833 (1988-10)
- [5] 岡田, 伊藤, 牧野, 城戸: 「構文駆動型連続DP法による連続音声中からの活用語のスポッティング」, 信学論(D),J70-D,12 (1987-12)
- [6] 岡田美智男: 「アクティブチャート解析法に基づく One-Pass アルゴリズムの構文制御について」, 信学技報 SP90-24 (1990-6)
- [7] T.Gunji: "Japanese Phrase Structure Grammar",Reidel (1987)
- [8] 伊藤, 小川, 牧野, 城戸: 「日本語 Dictation System における文節構造モデル」, 信学会技術報告 SP87-104, pp43-48 (1987-12)
- [9] 伊藤, 牧野, 城戸: 「CYK法に基づく統語処理アルゴリズム」, 音講論 3-3-12 (1988-10)
- [10] M. Tomita:"An Efficient Word Lattice Parsing Algorithm for Continuous Speech Recognition", Proc. ICASSP86 pp.1569-1572 (1986)

音声インターフェイスをもつ情報検索システムと談話処理

An Information Retrieval System with a Speech Interface and its Dialogue Processing

小林 豊 ・ 田辺 雅則 ・ 新美 康永

Yutaka KOBAYSHI, Masanori TANABE and Yasuhisa NIIMI

京都工芸繊維大学

Kyoto Institute of Technology

1 はじめに

情報機器との連続音声による対話のための研究はこれまであまり多くなかったが、近年、計算機の処理能力が飛躍的に向上してきており、実時間に近い処理速度をもつ音声対話システム実現に向けて研究が活性化されているといっている。

連続音声の中の音は前後音の影響を強く受けてその音響的な特徴は多様に変化し、単語境界の検出も困難なため、構文・意味・プラグマティクスなどの言語知識の助けを借りて入力音声と照合すべき単語列の候補をまず絞り込むことで処理労力を著しく削減することなしには連続音声の認識は実現できない。その上、音声対話に顕著な言い直しや冗長表現などの言語現象、指示語、省略の解釈などに加えて、文法的に正しくない文の入力を受理する柔軟さが、音声対話システムには要求される。さらに、従来開発されてきたテキスト入力の対話システムで強調されてきたように、ユーザに協力的な応答の生成が不可欠である。

我々は、人間と機械の音声による対話の実現を目指して、とくに目的指向型対話の一つの形態として情報検索システムの自然言語による質問応答を想定し、これを音声入力で利用するという立場からシステム構築を行ってきた [1-4]。本論文では、まず、発話の文脈を利用せずに、データベース検索コマンドに容易に変換できるような限定された構文の日本語文音声を入力とする情報検索システムを構築したので、音声認識部の性能と言語処理の方式について報告する。つぎに、ユーザがシステムと対話して都市の観光プランを作成していくような状況を想定した、より柔軟な対話を扱うことのできる談話処理部を含むシステムの構築について報告する。

以下、2. では、音声による情報検索システムの概要を述べ、3.~5. では、そのコンポーネントである連続音声認識システムの概要とその性能評価、タスクの構文的・意味的制約、認識文をデータベース検索コマンドに変換する処理と応答文の生成についてそれぞれ述べる。6. では、観光プラン作り支援のための対話システムの談話処理について述べる。

2 情報検索システムの概要

音声インターフェイスをもつ情報検索システムの処理の流れは、図1に示すように、ユーザが自然言語による検索文を自然な速度で発話した文音声を入力し、音声認識システムで

認識されると、その結果はコマンド変換部に送られて、データベース検索コマンドが生成される。情報検索の結果は応答文生成部に送られ、応答文テキストが作られる。

連続音声認識システムでは、構文的・意味的に許される単語列のうち、入力音声の音響パラメータ列に最もよく適合するものを探索法によって求めて認識結果とする。コマンド生成部・応答文生成部では、入力文の意味表現としてデータベースの検索コマンドに容易に変換可能な4字組を基本要素とする表現を用いている。

関係データベースは図2の例のような形式をもついくつかの表からなっており、各表には固有の名前があり、表の各行には属性値の組が1つのレコードとして書かれている。実際、特定のデータベースシステムを利用しているわけではなく、Prologで簡単なものを模擬している。現在のデータベースには、京都市の観光名所、遊園地、公共機関、ホテルなどの情報をもつ。

3 連続音声認識システム

3.1 システムの構成

連続発声された文中の音声信号が被る文脈による平滑化、すなわち、調音結合による音の変形は、連続音声認識システムを実現する上で最大の障害と言ってよい。従来から、階層モデルに基づいて構成される連続音声認識システムでは、トップダウンに単語予測を行い、ボトムアップに処理する音響分析結果と単語単位で検証することが多いが、調音結合は語中、語間を問わず起こるので、日本語のように助詞や助動詞などの短い単語が多く含まれる言語を認識対象とすると、単語を単位として認識・照合を行う方法は得策とは言えない。

そこで、本研究で開発した連続音声認識システム SUSKIT-II では、言語情報を用いたトップダウン処理は単語単位で行い、予測される単語列の発音テンプレートを一旦連結した後、音響分析部が安定に検出可能な音韻 (robust phone; 以後、RP と呼ぶ) のところで単語の単位とは関係なく分割して照合する方式を採用する。これによって、単語間にまたがる調音結合も単語中の場合と同様に扱うことができ、助詞などの短い機能語の多い日本語音声単語単位で照合する場合に遭遇する認識誤りの軽減を図ることができると期待される。

図3に SUSKIT-II の構成を示す。

音響分析部 — 10msec ごとに計算される LPC ケプストラム係数と短時間エネルギーおよびそれらの時間変化を計算し、それぞれコードブックサイズ 256 でベクトル量子化する。一方、日本語音素を 18 のクラスに分け、各音素クラスを 1 つの離散的 Hidden Markov Model (HMM) でモデル化して、訓練データを用いて HMM のパラメータを学習しておく。

単語予測部 — 制御部の選択した単語列の後に続く単語を構文解析と意味解析によって予測する。単語予測のアルゴリズムは、ボトムアップパーザ [5] を、多数の文仮説を並行して予測・検証していく必要のある連続音声認識システム用に改良したものであり、

DCG の規則から機械的な手続きによって単語予測用の Prolog のプログラムを導出する手順を先に報告した [6]。

音韻変形部 — 単語の音形はローマ字で記述され、音形規則を適用してラティス形式の単語発音テンプレートをあらかじめ作成しておく。音韻変形部では単語テンプレートの接続と単語間にまたがる音形規則の適用を行う。

音素列照合部 — 制御部から与えられた照合可能な部分音素列と音響分析部からの出力であるベクトル量子化されたコード系列とを比較する。トップダウン的に予測された部分音素列中の各音素クラスを対応する音素 HMM で置き換えた連結 HMM ネットワークを構成し、連結 HMM ネットワークがそのコード列を生成する確率を計算し、照合得点として制御部へ返す。

制御部（音素列照合スケジューラ） — 言語情報を用いたトップダウン予測は単語を単位として行い、音声データと照合は単語境界とは無関係の RP で囲まれた区間を単位として行う。このため認識の途中経過を 3 階層の木構造（品詞木、単語木、音素木）で記憶している。それぞれの節点は、品詞、単語および部分音素列に対応し、音素木の照合済み節点の優先順位にしたがって未照合部分がスケジューリングされ照合される。本研究では、left-to-right 処理によるビーム探索法を用いて時間同期で認識を進めて行く。

3.2 連続音声認識システムの性能評価

上述の音声認識システムの動作を確認するために、語彙数 248 の簡単なタスクで文認識実験を行った。

成人男性 3 名の発話した各 53 文を簡易無響室で録音した合計約 9.5 分の音声データを分析し、各人の発話を前・後半に分け、前半を用いて訓練した音素 HMM を用いて後半の文認識実験を行い、つぎに前後半を入れ換えて同様に文認識実験を行った。表 1 に示すように、話者によって 86.8%~94.2% の比較的高い文認識率を得た。また、認識された文の単語列はコマンド変換部へ送られ検索されて、期待通りの日本語による応答文テキストが、5. で述べるような方法で生成された。いまのところ音声応答モジュールは組み込まれていない。

4 タスクの言語的制約

4.1 文法規則

入力文の仕様は、文脈自由文法による構文規則と、意味マーカと格文法による意味制約を統一的に Definite Clause Grammar(DCG) で表現して用いている。本システムで採用した文法規則の中で用いる文法カテゴリーの分類は、関係表やその属性と密接に関連しており、以下の通り。

cp — 関係表の中のレコードのキーを指示する名詞句（「金閣寺」、「400 円で入ることのできる博物館」など）。

tp — 関係表中の属性名を指示する名詞句（「動物園の入場料」、「金閣寺が立てられた年」など）。

ppc, ppt (cp と tp に対応) — 動詞を含む述語句で、文を完結させるか、連体修飾句として用いられる。

sp — ppc における主題を表わす名詞句。

cfp — 比較の対象を表わす句（「三百円より（安い）」、「苔寺と（同じ）」など）。

dp — ppc や ppt の中の格を示す名詞句（「千年より前に建てられた」（時間格）など）。

一方、終端記号に相当する単語の文法カテゴリーは、名詞に関してはタスクに依存した意味的な細分類を以下のように行っているが、他の品詞に関しては概ね通常の日本語文法に準拠した分類に従っている。

cn — データベース中の特定の表を指示する名詞（「社寺」、「大学」など）。

tn — 関係表中の属性名を示す名詞（「拝観料」、「電話番号」など）。

cst — 関係表中の「名前」という属性欄に現われる名詞（「銀閣寺」、「工繊大」など）。

tcst — 関係表中の「名前」以外の属性欄に現われる名詞（「百円」、「十時」など）。

qn — 疑問詞（「何」、「誰」、「何処」など）。

4.2 意味規則

意味マーカ： 単語間の共起関係を記述するために意味マーカの役割を果たすパラメータを使用している。われわれのタスクにおける名詞と名詞の間の意味的制約は、(a) 関係表の名前欄と属性名の間「AのB」という関係、(b) 「CはD」のように属性名とその属性値、あるいは属性値同志の比較などの際に要求される意味的整合関係の2つに大別することができる。

これらの意味的な制約を表現するのに、2種類の意味マーカ C と T を使用している。C は文法カテゴリー cn と cst に属する名詞にパラメータとして与えられ、関係表の種類を値として持つ。T は文法カテゴリー tn と qn に属する名詞に与えられ、属性名の上位概念（時間、場所、価格など）を値としてとる。文法カテゴリー tcst に属する名詞は両方の意味マーカをパラメータとして持つ。文法規則中には、意味的制約 {cat(T,C)} を適宜挿入して T が C で指定された関係表の属性であることを示している。

格構造： 「十時より早く」、「五百円より安い」といった句における形容詞とこれにかかる修飾句中の関係は、形容詞にも意味マーカを与え、さらに助詞の意味マーカ F との共起関係を一種の格構造の形で記述しておく。文法規則中には意味的拘束条件を判定する述語 {comp(T,F)} を埋め込む。

動詞の意味記述は格文法的な考え方に従って、文法規則中では {verbf(T,F,C,V)} で格構造の制約を記述する。V は動詞のタイプを表わす。

3.2 で報告した文認識実験に用いた文法の語彙数は 248 であったが、その後、品詞分類を改良するとともに、5. で述べる通り多様な検索が行なえるように、許される文型を増やしたので、現在、語彙数は 306 である。

5 コマンド変換と応答文生成

5.1 コマンド変換部の概略

音声認識部は認識結果として、構文・意味的に制約を満たす単語列で入力音声に最もよく照合するものをコマンド変換部（以後、単に変換部と呼ぶ）に渡す。変換部では、この単語列をデータベース固有の知識を用いて再び解析して、検索コマンドに変換する。

入力文の意味表現である内部コードは、タスクの目的に応じた入力文の意味解析を与えると共に、情報検索システムのコマンドへの変換が容易になるような形式が望ましい。現在のシステムでは特定の情報検索システムを想定しているわけではないが、例えば、関係データベース検索言語 SEQUEL-2 などのコマンドに容易に変換できる形式になっており、実際には、Prolog のプログラムで情報検索システムを模擬しているので、この内部コードを一種の検索コマンドとして解釈している。

検索文は一般に図 4 に示す 2 つの形式に大別できる。図 4(a) に対する検索の結果は関係表中のあるレコードのある属性の値が返され、また、図 4(b) に対する回答としては、yes/no が返される。応答文生成部ではこの検索結果を入力し、入力検索文の内部表現を参照して、応答文を生成する。応答文はテキスト出力として、音声インターフェイスの利用者に提示される。

5.2 意味表現の形式

データベース検索のための意味表現としての内部コードの構成規則は図 5 に示すものを採用している。ここで、演算子とは、検索部での行動を決定するための情報で、図 4 に示した質問文の形式に対応して 2 種類に分類される。図 4(a) は疑問詞を伴っており、データベースの表からある値を取り出すことを要求し、図 4(b) では文中に指定された値が正しいか誤っているかの判断を要求している。それぞれに対応する演算子は、find（値を取り出せ）、==（一致）である。このほか、演算子としては、不一致、大小比較に加えて、新たに min、max、nearest を設けるとともに、複数の項目の検索も許して、次の例のような入力を受理できるように改良した。

一番早く開く博物館はどこですか。

左京区にある大学の名前と電話番号を教えてください。

京都駅に一番近いホテルはどこですか。

4 字組には次の情報が含まれる: (a) データベースの表を指示する情報 (C)、(b) 表の中のレコードを指示する情報 (N)、(c) 属性名を指示する情報 (T)、(d) 属性値 (V)。これらの内の 3 つを入力検索文より抽出して、残りの 1 つを検索できる構造に仕上げるのである。図 5 に示す 4 字組の構文規則からわかるように、N および V の位置に 4 字組を使用することができる。これは、検索文が連体修飾などの埋め込み文を含み得ることと対応している。

5.3 意味解析

入力文の意味解析は構文解析と並行して行われる。各単語の意味の定義は、演算子と 4 つの情報の一部しか値の定まっていない不完全な 4 字組の組み合わせとして与えられる。図 6 に意味定義の一部を示す。

また、各文法カテゴリーにも同様の形式の意味表現とそれを構成する規則が与えられている。意味定義の構成規則は構文規則に対応しており、構文規則の右辺にある各文法カテゴリーの意味表現から、左辺の文法カテゴリーに対応する意味表現を構成する方法を与える。構文解析の進行と並行して意味構成規則を適用して行くことにより、文の意味表現を得ることができる。図 7 に質問文「金閣寺の拝観料を教えてください」の構文・意味解析の過程を示す。

5.4 応答文の生成

情報検索の結果は、入力の意味表現の演算子が find の場合は完成した 4 字組が回答となり、比較演算子の場合は yes または no が返される。そこで、質問文の意味表現と応答文の形式を対応させる規則を用いて応答文を生成する。図 8 に応答文生成規則の一部を示す。図 7 の意味表現によって検索した結果が、500 円であったとすると 4 字組は「社寺, 金閣寺, 拝観料, 500 円」となるので、応答文の原型は「金閣寺の tn(料金, 社寺) は 500 円である」となり、辞書を引くことにより tn(料金, 社寺) のところに「拝観料」が代入されて応答文が完成する。

6 音声対話のための談話システム

前節までは、文脈情報を利用しない日本語情報検索文の音声認識と応答生成について述べた。現在、ある都市を 1 日ないし数日間観光するプラン作りのためにユーザがシステムに相談するような状況における対話を考えて、談話システムを構築しており、音声インターフェイスを結合した音声対話システムに発展させるため研究を進めている。

談話システムは、談話目標抽出を含む構文・意味解析部、文脈情報の履歴を解析する対話構造解析部、情報検索・プラン作成部、システム主導型およびユーザ主導型の対話モードを取り扱う対話制御部、応答生成部からなり、現在、テキスト入出力で動作する。音声認識システムとの結合においては、単に認識結果としての単語列を受けとるだけではなく、音声認識システムの多大な照合労力を軽減するため、談話解析によって次発話に現れ得る単語や構文を予測限定することは非常に重要であり、音声合成システムに対しては、応答文の文法構造、強調すべき構成要素などを送る。このために、我々は先に、話題管理方式を提案しキーワード予測の予備実験の結果を示した [4]。この節では談話システムの知識表現、談話目標遷移モデルに基づく対話制御および動作例について報告する。

6.1 文脈情報の表現と管理

我々のシステムでは、システムが対話の主導権をもって、観光計画をたてるという大きな目標に向かって利用者の意向を明確にしていく動作モードと、利用者が主導権をもって質問をしながら、システムから情報を引き出したり、システムが提示した観光地をプランに含めるかなどを判断していく動作モードがある。

このとき両者の間で行なわれた質問応答の結果を話題の推移と談話目標という2つの観点からとらえ、これを文脈情報として管理しておく。話題の推移は、システムの提示した複数個の話題を OR 節点に、利用者の提示した複数個の話題を AND 節点に対応させた AND-OR 木の形で表現できる。これを「話題木」と呼ぶことにする。ただし、節点の性格付けは動的なもので、話題の進行と共に変化することがある。

一方、談話システムは観光案内に関する関係データベースをもっていて、ユーザとシステムはそれを検索しながら対話を進行させて行くものと考えている。そこでは、大まかな観光プランから始めて詳細化していく過程は、上位の関係表から次第に下位の関係表に移るように話題を展開していくことと対応するので、データベースの関係表の名前、各関係表に含まれる属性名とそれぞれのとり得る属性値が話題の静的な階層構造をなすと考えることができる。これを「話題の階層構造」と呼ぶ。

また、我々は、多くの対話例を解析することによって、個々の発話のもつ談話目標を解析し、次のように分類した。

システムの談話目標 — 話題提示、条件質問、候補提示、知識回答、確認質問

ユーザの談話目標 — 条件提示、条件回答、候補要求、別候補要求、知識要求、確認回答、受諾、終了回答

談話目標の遷移規則の一部を図 9 に示す。システム発話は矩形で囲まれており、受諾の発話の後の遷移などは、話題木の AND 節点や OR 節点の話題がすべて選択されたかどうかなどに影響される。

6.2 談話構造の解析と談話フレーム

入力発話は構文・意味解析部で意味表現に変換され、対話制御部に送られる。ここで話題の候補となる単語をその優先順位とともに取り出す。主題格、対象格、目的格になっている名詞句やサ変動詞の語幹などが高い優先順位をもつ。つぎに発話の意味表現から選ばれる談話目標の候補のうち、談話木と談話の階層構造に照らして適合するものを談話目標と定める。

1つの発話の内容を表すためには図10のようなフレームを用いる。「意志」には文のタイプが、「述語」にはユーザ発話の動詞が入る。「話題」にはユーザ発話で提示された話題とその上位概念が、「焦点」には話題の階層構造の中で着目されている話題の名前が、「可能」には話題木のAND節点やOR節点でユーザが未選択の話題のリストが代入される。最後の4つはまとめて「表関係スロット」と呼び、ユーザ発話の解析の結果代入されて、具体的な知識要求の内容を明らかにし応答を生成するために利用される。前節のコマンド生成の場合と異なり、発話の中で省略された項目があると話題木から補ってスロットを埋めてから、検索コマンドを生成する。応答生成は基本的には前節で説明したとほぼ同様な方法を用いるが、以下の手順に従う。

- (1) ユーザ談話目標を調べる。
- (2) ユーザ意志とユーザ述語から応答のための述語（動詞）を決定する。
- (3) 焦点、話題、可能、表関係から提示すべき内容を決める。このとき、話題の階層構造も考慮する。
- (4) システムが応答した談話目標スロットを埋める。
- (5) システムの提示した内容を話題木に加え、談話目標遷移規則に従ってシステム主導あるいはユーザ主導の制御モジュールに制御を渡す。

音声認識システムに渡す予測キーワードを決めるには、話題木や話題の階層構造を参照してスロットが埋められた現在の談話フレームと談話目標遷移規則を参照して、次発話に含まれ得る話題を予測限定することにより行なう。

ユーザがテキスト入力によりシステムと対話した実例を付録に示す。

‘S:’と‘U:’は、それぞれ、システム発話とユーザ発話を表しており、‘*** frame ***’に続く行に‘[何, 竜安寺, 庭, 特徴]’などとなっているのは、発話から抽出された話題の候補である。それに続いて、ユーザ発話を解析して作られる談話フレームの内容が表示されている。ただし、値が確定したものだけ表示してある。

システムの最初の発話は固定されているが、2番目以降のシステム発話は、一連のユーザ発話の解析と応答生成規則に従って自動生成されたもので、所期の動作が確認された。語彙数は630単語、解析のための文法規則数は77であった。

7 まとめ

音声インターフェイスをもつ情報検索システムの仕様、連続音声認識部 SUSKIT-II の構成と性能、情報検索コマンドと応答文の生成方法について報告した。SUSKIT-II では、3 名の話者の発声した 53 文に対する文認識率は 86.8%~94.2%を達成した。情報検索のためのコマンド生成と応答生成部では、データベースの構造を反映した 4 字組に基づくモデルを提案し、受理可能な検索文の構文を拡張し、オンラインシステムとして実現した。

一方、観光プラン作りをタスクとする音声対話システムの構成と談話処理の機構について報告した。今後、より柔軟な対話が実現するように、システムの知識表現と言語処理方式を精密化するとともに、音声インターフェースとも結合して性能評価を行ないたいと考えている。

参考文献

- [1] 新美, 小林, 米沢: “対話システムにおける対話構造の解析”, 重点領域研究「音声言語」研究報告, PASL89-4-2 (1989).
- [2] 小林, 遠藤, 新美: “音声インターフェイスをもつ情報検索システム”, 信学技報, SP90-41 (1990.8) 研究報告, PASL01-4-5 (1989.10).
- [3] 田辺, 小林, 新美: “連続音声認識システム SUSKIT-2 の高速化”, 信学技報, SP91-25 (1991.6).
- [4] 新美, 小林: “対話音声理解システムにおける話題管理とキーワード予測”, 信学技報, SP87-103 (1987).
- [5] Y. Matsumoto, et al: “BUP: a bottom-up parser embedded in Prolog”, New Generation Computing, 1, 2 (1983).
- [6] 新美, 小林, 渦原: “音声理解システムにおける言語情報のトップダウン的利用方式”, 信学論誌 D, J70-D, 9 (1987.9).

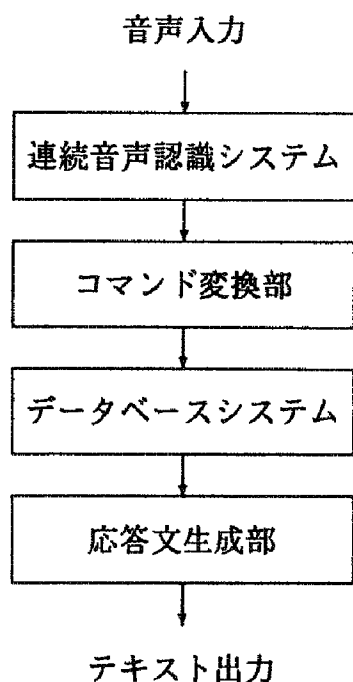


図 1. 音声インターフェースをもつ情報検索システム

関係表：神社仏閣

名 前	場所	拝観料	開館時刻	
金閣寺	衣笠	400	9:00	
銀閣寺	東山	400	8:30	
鹿王院	太秦	300	9:00	

図 2. データベースの形式

表 1. HMM照合による文認識実験結果

話者	文認識率 (%)	
	1 位	～ 5 位
KB	86.8	96.2
NY	94.2	98.1
HA	88.7	88.7
平均	89.9	94.3

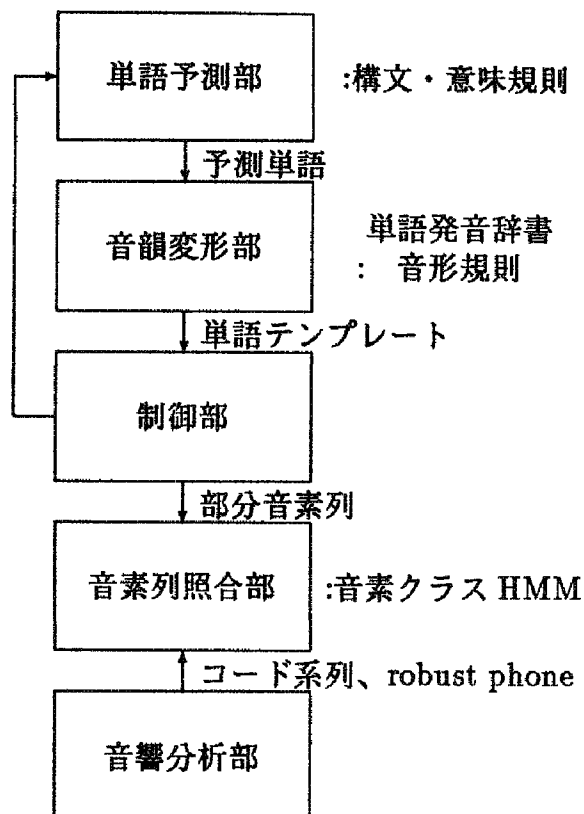


図 3. SUSKIT-II の構成

- (a) 金閣寺の拝観料はいくらですか。
 (b) 金閣寺の拝観料は五百円ですか。

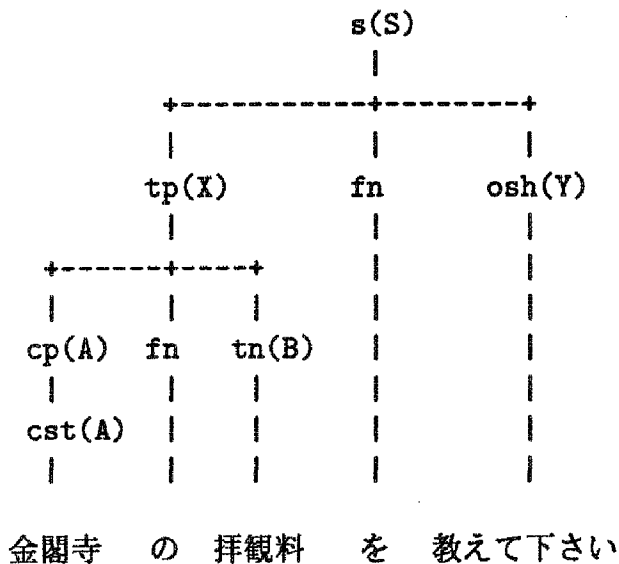
図 4. 検索文の 2 つの形式

文の意味表現 → {演算子, 4 字組}
 | {演算子, 4 字組, 4 字組}
 4 字組 → [C, N, T, V]
 C → SM1
 T → SM2
 N → SM1 | 4 字組
 V → SM2 | 4 字組 | 比較式
 比較式 → (比較演算子, 変数 1, 変数 2)
 演算子 → find | 比較演算子
 比較演算子 → = | == | =< | < | > | >=
 | /== | min | max | nearest
 SM1 → 意味マーカ C の値 | 空白
 SM2 → 意味マーカ T の値 | 空白

図 5. 文の意味表現の構文

cn → [C, -, -, -] (社寺 → [社寺, -, -, -])
 tn → [-, -, T, -] (拝観料 → [-, -, 料金, -])
 cst → [C, N, -, -] (金閣寺 → [社寺, 金閣寺, -, -])
 adj → {op, T} (安い → {>, 料金})

図 6. 単語クラスとその意味表現



意味表現

A = [社寺, 金閣寺, -]
 B = [社寺, -, 料金, -]
 X = [社寺, 金閣寺, 料金, -]
 Y = find
 S = {find, [社寺, 金閣寺, 料金]}

図 7. 意味解析の例

- (1) {find, [C, N, T, V]} (V が未知のとき)
「N の tn(T, C) は V である」
- (2) {find, [C, N, T, V]} (N が未知のとき)
「tn(T, C) が V の C は N である」
- (3) {op, [C1, N1, T1, V1], [C2, N2, T2, V2]}
(op が =, <, >, >= のとき)
「N1 の tn(T1, C1) は N2
の tn(T2, C2) | V2 より adj(op, T) で
す (ではありません)」

図 8. 応答文の生成規則 ({} 中の縦棒は 2 番目の 4 字組の形式に従ってどれか 1 つをとることを意味する)

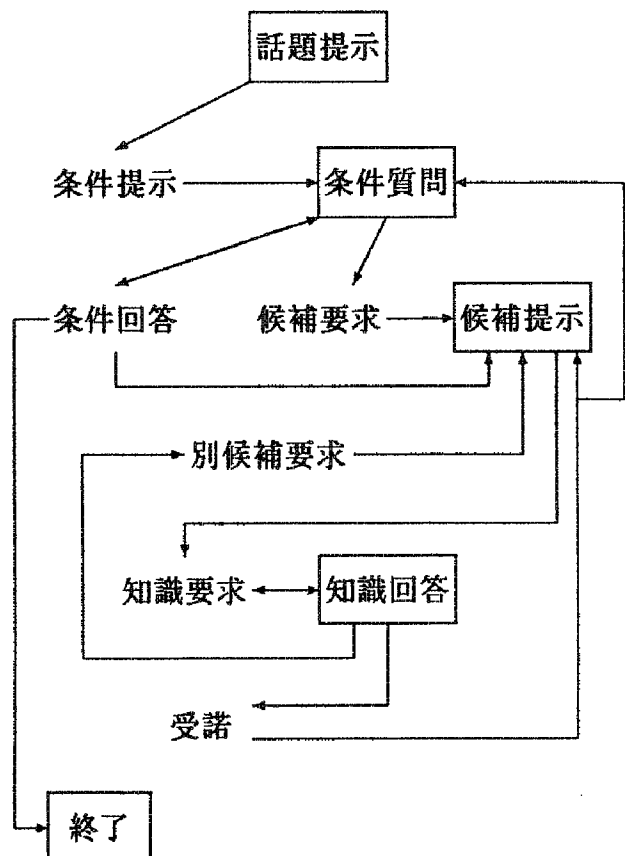


図 9. 談話目標遷移規則

ユーザ談話目標:

意志:

述語:

話題:

焦点:

話題階層構造中の root からの経路:

可能:

システム談話目標:

関係表の名前:

名前:

属性名:

属性値:

図 10. 談話フレーム

付録

S : 京都観光案内システムです
観光プラン作りをお助けします

U : 2、3日観光したいのですが

*** frame ***

[2、3日, 観光, の]

ユーザー談話目標 (条件提示)

意志 (婉曲要求文)

述語 (観光する), 話題 (日程), 現在 (観光)

焦点 (日程)

システム談話目標 (条件質問)

S : 何に興味がありますか

庭、伝統産業などがありますか

U : 庭園です

*** frame ***

[庭園]

ユーザー談話目標 (条件回答)

意志 (平叙文),

述語 (です), 話題 (庭), 現在 (興味),

焦点 (庭), 表名 (庭)

システム談話目標 (条件質問)

S : どんな時代の庭がいいですか

桃山時代、江戸時代、室町時代など
ありますが....

U : 室町、桃山、江戸の名園が見たいと思
います

*** frame ***

[江戸, 桃山, 名園, 室町]

ユーザー談話目標 (条件回答)

意志 (平常文),

述語 (見る), 話題 ([時代, 室町, 桃山, 江戸]),

現在 ([観光, 興味, 庭]), 焦点 (時代),

表名 (庭), 属性 (時代), 属性値 (室町時代),

属性値 (桃山時代), 属性値 (江戸時代),

システム談話目標 (候補提示)

S : まず最初に、室町時代の庭は金閣寺、銀
閣寺、竜安寺です

U : 竜安寺の庭の特徴は何ですか

*** frame ***

[何, 竜安寺, 庭, 特徴]

ユーザー談話目標 (知識要求)

意志 (質問文),

述語 (です), 話題 ([特徴]),

現在 ([観光, 興味, 庭, 時代, 竜安寺]),

焦点 (特徴)

可能 ([銀閣寺, 金閣寺, 桃山, 江戸]),

表名 (庭), 名前 (竜安寺), 属性 (特徴),

システム談話目標 (知識回答)

S : 竜安寺の特徴は、いくつかの石と砂だけ
で作られていることです

U : 作者は誰ですか

*** frame ***

[誰, 作者]

ユーザー談話目標 (知識要求)

意志 (質問文),

述語 (です), 話題 ([作者]),

現在 ([観光, 興味, 庭, 時代, 竜安寺]),

焦点 (作者),

可能 ([銀閣寺, 金閣寺, 桃山, 江戸,

伝統産業]),

表名 (庭), 名前 (竜安寺), 属性 (作者),

システム談話目標 (知識回答)

S : 竜安寺の作者は、分らないです

U : では、寺は誰が建てたのですか

*** frame ***

[建立者, の, 誰, 寺]

ユーザー談話目標 (知識要求)

意志 (質問文),

述語 (建てる), 話題 ([竜安寺, 建立者]),

現在 ([観光, 興味, 庭, 時代, 竜安寺]),

焦点 (竜安寺),

可能 ([銀閣寺, 金閣寺, 桃山, 江戸,

伝統産業]),

表名 (社寺), 名前 (竜安寺), 属性 (建立者),

システム談話目標 (知識回答)

S : 竜安寺の建立者は、細川勝元です

*** end ***

口語調音声による 情報検索システムの試み

市川 薫 天野 明雄 北原 義典

(日立製作所 中央研究所)

1. はじめに

音声入力の実用形態には様々なものが考えられる。その中で、音声の利点が最も活きるものは、対話的な使い方(意図の入力/伝達)であろう。音声タイプライタ等データ入力は、実験を行ってみると、書き言葉としての正しい文章をいきなり直接音声で入力することは普通の人には困難、長時間の発声は疲れる、入力内容を周囲の人に聞かれるという心理的抵抗感、等があり向いていないように思われる。ヒューマン・インタフェースとして意味の在る応用形態とは何かを追及し、それを実現するうえで真に重要な技術の開発を図るべきである。情報の入出力には、それぞれの情報の種類や性質に応じて適切なメディアを用いるべきであり、不向きなものへの無理な適用は、その後の効果的な利用に対しても利用者に不信感を植え付ける原因となろう。

ヒューマン・インタフェースとして音声の利点が活きるためには、入力に気を使わない(利用者の思考中断を引き起こさない)ように、自由な発声の許容と実時間性が実現されなければならない。これには、非常に多くの困難な課題を解決しなければならないことは、既に多くの研究者が指摘している通りである。

我々は、自由発声音声の入力を許容する技術の開発を目指して、これまでに対話音声の実態を観察、その結果に基づいて様々な技術を提案し、その可能性について基礎的な検討を加えてきた。抑揚情報を重視した一連の検討[Komatsu1990など]や、音声の性質(不完全な表現が多い)を前提とした理解方式の検討[Komatsu1986]、曖昧な発声の音韻認識手法[Amano1989]、意味情報の利用の検討[Hataoka 1989]、対話入出力間の能動的処理の提案[Komatsu 1991]等である。

一方、最近では、例えば、隠れマルコフモデルのように、音声認識技術にも様々な発展が見られる。また、情報処理を支えるハードウェアの性能向上も目を見張るものがある。工学的立場からは、これら周囲の状況の変化に注目し、要素技術の検討だけでなく、実現性の高い技術を組合せることによってシステムを纏めることも重要である。もちろん、この場合、利用者にとっての音声をもちいる利点を維持することを前提としなければならない。

そこで今回は、現時点で実現可能な技術を組み合わせたシステムがどの程度のものになるのかの検討と、音声対話の問題点を実験的に検討するツールとしての実験システムを試作した[村山1991][北原1991]。

ここでのシステムの視点としては、次の様な考え方をとった。

(1) 人間のように多様なタスクを扱う必要はない。タスクは限定しても、そのかわりに、発声には出来

るだけ無理な制限を付けないことが望ましい。

(2) 対話シーケンスを完全に自由にすることは、必ずしも使い勝手を良くしない。有る程度システムがガイドする方が迷わずにすむ。しかし、その流れを利用者の意志によってある程度変更したり、修正出来るようにしたい。

(3) タスクを限定し、対話の流れをシステム主導とすることで、テニヲハ等の曖昧な発声が正確に認識出来なくとも、文脈情報の利用が可能となりキーワードがわかれば、利用者の意図の理解がほぼ可能となる。

(4) 人間の対話に於いても各発声内容が100%理解されているわけではない。対話の利点を活かして聞き直しや訂正機能を持たせることによって、効果的なシステムが期待出来る。

(5) 実際の対話では、長文や複雑な構造を持った文が発声されることはほとんど無い点に注目する。

(6) 強力なアルゴリズムが存在し、性能の高いHMM方式をベースとしたい。

(7) 正規文法を用いる。正規文法は、生成文法の中では最も制約の強い文法であるが、修正、追加等更新が容易であり、本検索システムのような限定タスク下で比較的シンプルな文を扱う条件のもとでは有効なものと考えられる。

(8) 実時間性は思考の流れを中断させないために不可欠である。

(9) 試作システムは様々な条件での評価実験や改良にも試用できるよう、汎用性の高い機器構成とした。

タスクとしては、情報検索システムを選んだ。利用者と機械の間で相補的な関係にある利点の組合せによる効果を期待出来るアプリケーションの一つと思われる。すなわち、将来本格的音声対話が実現されれば、

A. 機械の得意とする機能

a. 大量なデータを正確に保持、検索できる。

b. 多段な論理を正確に実行できる。

など。

B. 人間の得意とする働き

a. 直感による判断が得意。

b. 感性に優れている。

など。

のA. aの機能を(B. aを損なわずに)活かすシステムとなるものと期待される。

2. システムの概要

検索の具体的対象としては、建築およびインテリア情報を選び、対応するカラー写真を表示出力するシステムを試作した。建築を計画している人が自分の希望する建物、インテリア等の条件を音声で入力し、検索条件に合致するサンプルのカラー写真を検索出力して計画立案の参考にする場面を想定している。システムには100枚の建築カラー写真が格納されている。

Fig. 1に概要を示す。システムは汎用ワークステーションとアレープロセッサからなる音声入力部と光ディスクを持った情報検索部から構成されている。音声認識部は正規文法から生成可能な文章をHMMのネットワークに展開し、処理を行っている。

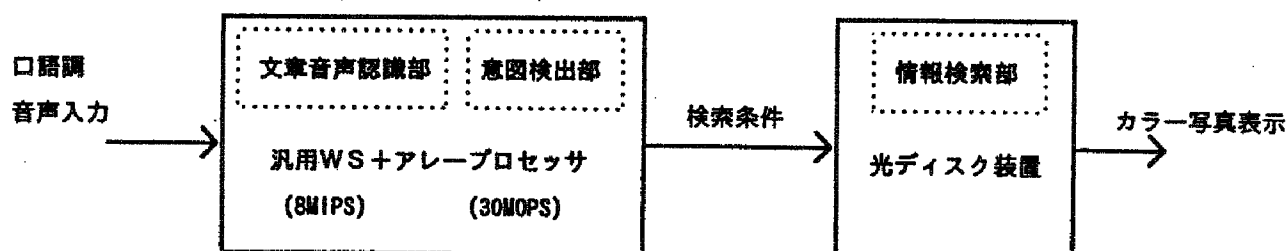


Fig. 1 試作システムの機器構成

3. 対話制御

一般に利用者は建築やインテリアの専門家とは限らない。従ってどのような検索条件を入力すれば良いかは判っていないのが普通である。そこで、こうした利用者をガイドするためにシステム主導の形態を基本とした。建物の持つ主な属性の分析を行い、重要と思われるものから順番にシステムがガイドし（システム主導）、利用者が各々の属性に該当する希望内容を入力するように構成した。

しかし、利用者にとってシステムの固定的ガイドに従うというのは決して使いよいものではない。入力を要求された検索条件について特に希望の無い場合や、一度入力した検索条件の訂正をしたい（音声認識の誤りもある）場合も考慮し、それらの状況により利用者の変更できるように構成した（ユーザ主導）。即ち、ガイドに従った検索条件の入力の他に、変更の為のシステムへのコマンド入力も常に受け付けることが出来る（簡易な主導権混在方式）。コマンド入力は、そのまま実行すると誤認識時に取り返しがつかなくなるため、実行するかキャンセルするかをプルダウン・メニューでを示し、音声で確認をする方式とした。

Fig. 2に建物の持つ主な属性と、その各属性に対する希望の条件（属性値と呼ぶ）の例を示す。また、Fig. 3に入力音声に従って対話の状態がどのように変化するかを示す状態課程の例を示す。太線はシステムのガイドに従って検索条件を音声入力したときの状態遷移を示し、細線はコマンドの音声入力による状態変化を示す。なお、検索条件をスキップする場合は、ガイドに対し「何でもかまいません」などと応えることによって、検索条件文として扱っているが、条件入力を打ち切ったり、入力条件を訂正するために前の状態に戻る場合は、コマンド文として扱っている。

ガイダンスの指示、音声コマンドに対するプルダウンメニュー表示などはMotifに準拠した。

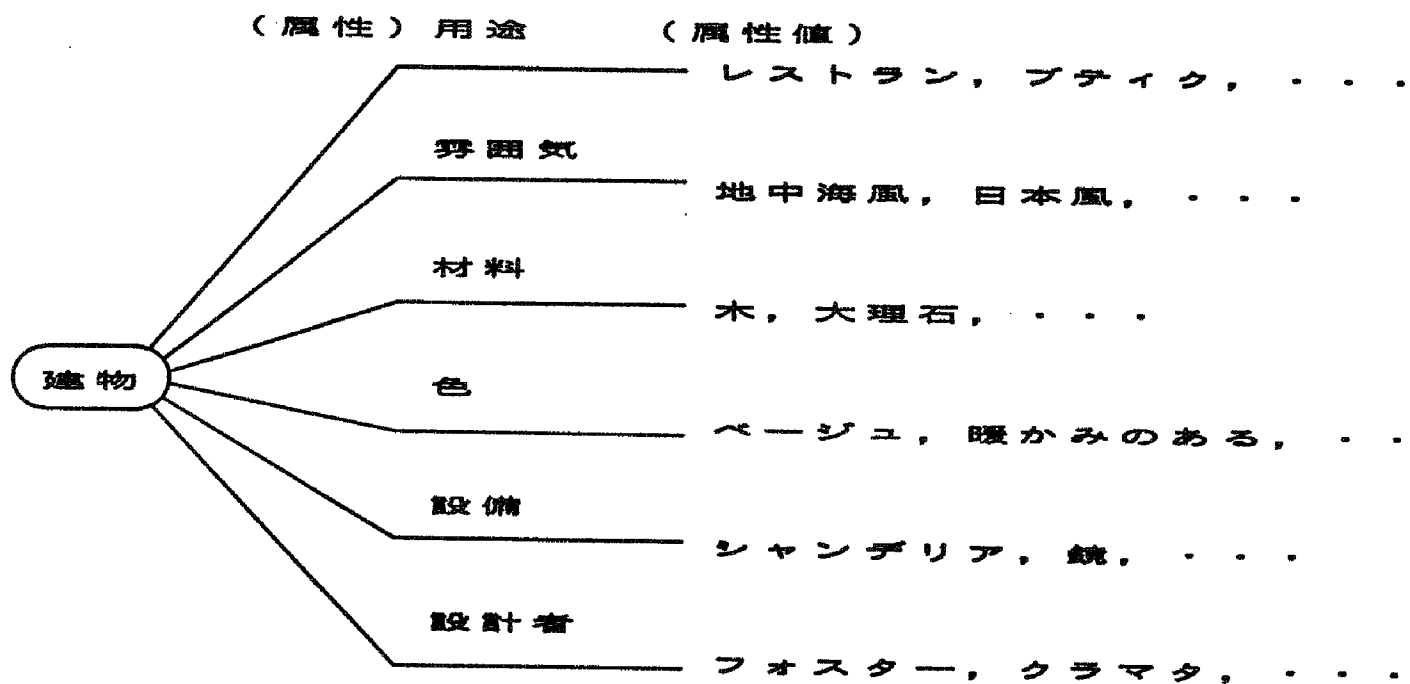


Fig. 2 建物の持つ属性と属性値

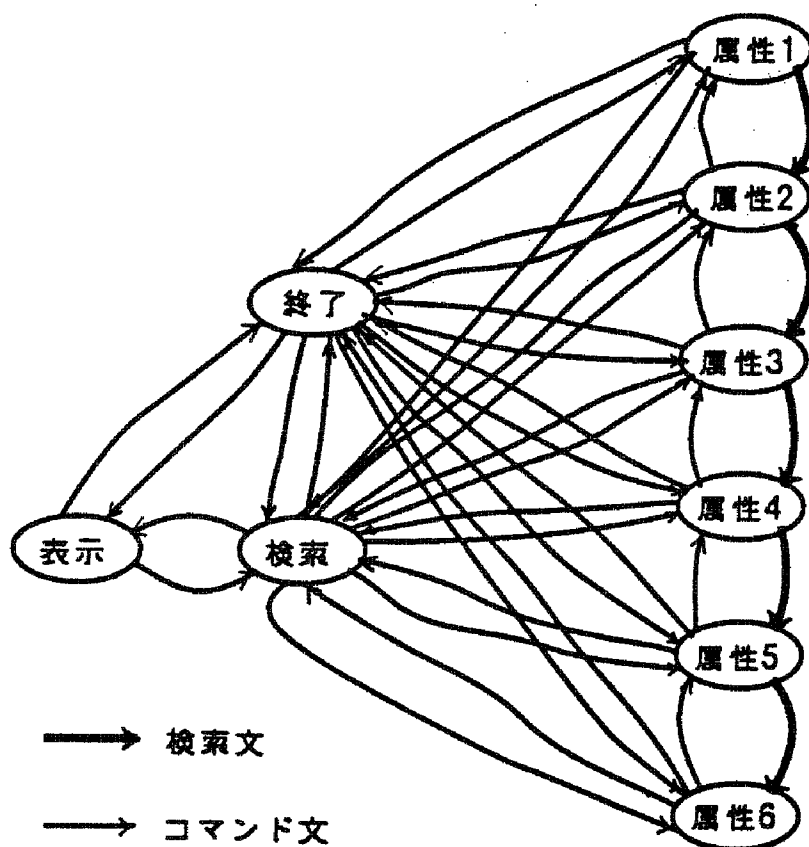


Fig. 3
検索システムの
状態遷移図

4. 意図の理解

正規文法中に注目概念（キーワード）スロットを埋め込んでおき、認識の結果選ばれたパス中の同スロット内の単語を同定することによって利用者の意図を理解する。キーワードの認識が正しければ、その他の部分に認識誤りが有っても、意図を理解することが出来ることが多い。キーワードは比較的丁寧に発声されることが多いので、単語の平均認識率より高い認識率が得られることが期待される。

本システムで使用する正規文法では、終端記号を単一の単語ではなく、単語の集合としている。ネットワーク・コンパイラは、先ず単語集合からなるネットワークに展開し、単語辞書を使用して更にその集合に属する単語からなるHMMのネットワークに展開する。Fig. 4にこの正規文法を示す有限状態オートマトンとそれを単語に展開した後のネットワークの例を示す。なお、注目概念スロットに対応する単語辞書を注目語辞書と呼び、その他のスロットに対応させる単語辞書を非注目語辞書と呼ぶ。

本システムでは、対話の進行に伴い、各時点で必要となる正規文法を選択し、認識のためのネットワークを各時点で動的に生成して用いている。

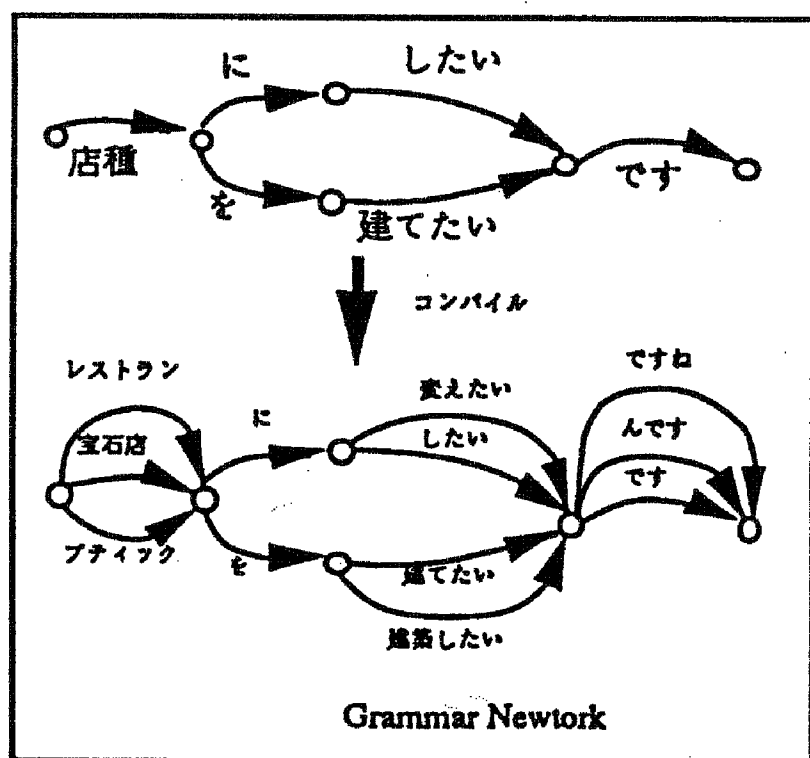


Fig. 4

ネットワークの展開例

5. 柔軟な口語表現への対応

本システムでは、語用論に基づく口語文生成ルールを構成し、本ルールによって生成された単語集合列よりなる文パターンを口語文受理正規文法として持たせることによって、柔軟な口語表現に対応している。

口語文生成ルールは、Fig. 5に示す2段階の口語文特性に基づいて構成したもので、単語集合の書換えルールである。同図に示す特性は、建物の6種類の属性に関する質問に答える口語文（1文）で、ユーザの“希望”を前提としており、7人の男女が発話した431文に基づいて整理したものである。同特性のうち、⑤「時制の変動」⑥「発話の中断」については例が少数であり、今回はルール化しなかった。本ルールの適用によって得られた口語文パターンによって“文型”の変動を吸収した。

③の「語の置換」は非注目語の単語集合に類義語として登録されている単語を用いて行われる。これにより「語」の変動を吸収している。

(0) 基本動詞(形容詞)の割当て

用途…建てる.
設備…付ける
…
共通…欲する、見る、いい

(1) 意図の移入

<動詞連用><願望助動詞「たい」>
<動詞未然><意志助動詞「よう」「う」>
<動詞連体><意志を表わす名詞「つもり」「予定」>等

(2) 構文の変形

①語の付加

- a. 複合辞
[<判断を表わす動詞「と思う」>] [<断定助動詞「(ん)だ」「(ん)です」>]
[<接続助詞「けれども」「が」>] [<親愛終助詞「よ」「ね」>] 等
- b. 間投詞 「あー」「あの」「うーん」「えー」「ええと」等
- c. 連用語 「やっぱり」「ちょっと」「絶対に」等
- d. 接続詞 「実は」「どちらかといえば」等
- e. 主題 「～としては」「～については」「～は」等

②語の省略

- a. 旧情報、既出語 「レストランを建てたいと思う」⇒「レストランをと思う」等
- b. 主語 「私はレストランを建てたい」⇒「レストランを建てたい」等
- c. 場所格 「天井にシャンデリアが欲しい」⇒「シャンデリアが欲しい」等

③語の置換

- a. 類義語 平易化: 「建てる」⇒「欲しい」「作る」等
短縮化: 「けれども」⇒「けれど」「けど」等
ぼかし: 「が」⇒「なんか」「とか」「の方が」等
- b. 照応 指示語: 「建物の色」⇒「その色」等

④語順入替え

- a. 場所格と対象格 「ピアノを客席に置きたい」⇒「客席にピアノを置きたい」等

⑤時制の変動

⑥発話の中断

Fig. 5 口語文生成ルール(検索文の場合)

本システムでは前述の様に「キャンセル」「検索開始」等のコマンドに関する文章も、柔軟な口語形式で入力が可能であり、各状態でコマンド文章と検索文のどちらも受け付けるように構成されている。

使用した総語彙数は345語、各属性ごとの同時認識可能な文章のタイプは、検索文とコマンドに関する文章の合計であり、平均367タイプ(max 1754, min 168)、これに各注目語辞書や非注目語辞書から具体的単語などをいれて展開した具体的文章の同時認識可能な数は、属性当り平均258, 718文(max 729, 582, min 24, 424)であった。ブランディングファクタは平均4.35、一文当りの平均単語連結数は10.7である。

6. 認識方式

認識の基本単位にはCVおよびVCVを使用している。単語辞書は、音韻記号列で記述されている。

認識に先立ち、情報検索部の状態情報（Fig. 3参照）に基づき使用する正規文法を選択する。本システムで使用する正規文法は、先に述べたように終端記号が特定の単語ではなく、単語の集合を表わす概念に対応させている。ネットワークコンパイラは、概念に属する単語を、辞書の情報を利用して展開し、文章のHMMのネットワークに展開する。HMMは単語内部では調音結合に対処するためにVCVを単位として用いているが、単語結合部分では処理量を削減するためにCVを用いている。

ネットワークサーチはViterbiアルゴリズムを基本として、処理量削減のためにビームサーチによっている。さらに高速化を図るためHMMの各状態を単位とした枝刈りを行っている。

認識の特徴パラメータとしては、LPCケプストラム、 Δ ケプストラム、パワー+ Δ パワーを用いた。

7. システム構成

以上に述べた各機能からなるシステム構成をFig. 6に示す。また、Fig. 7は試作システムの検索画面である。

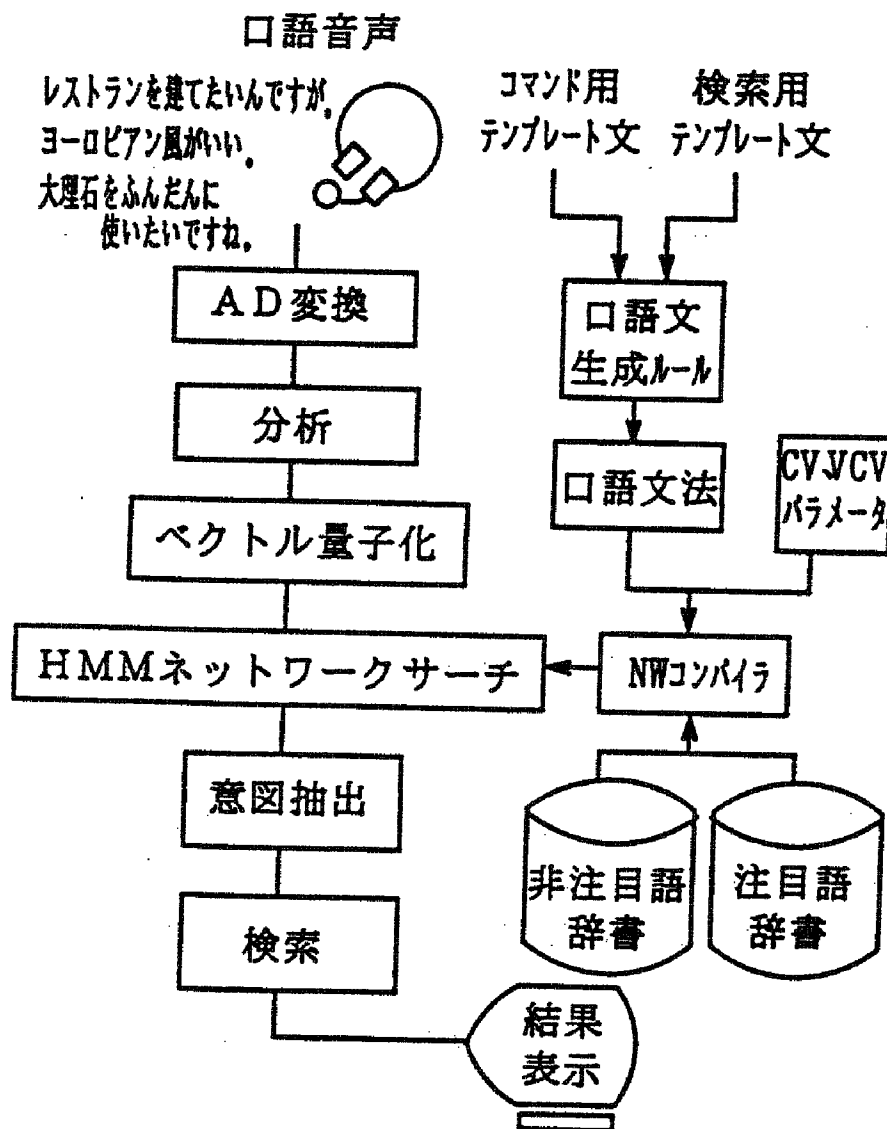


Fig. 6

口語調音声による
情報検索システム構成

中段の「メッセージ」欄に現れるガイド文に従って音声を発声すると認識結果が下段のウィンドウの中に表示され、認識結果の文中の注目概念に対応する単語をキーワードとする検索条件が左上のウィンドウに表示される。検索コマンドの文章を音声で入力すると、システムはプルダウンメニューを表示してコマンド認識結果を示し、利用者が実行を音声で指示すると検索が行われ、その結果検索条件の一致したものの件数とデータの番号が右上のウィンドウに表示される。更に画像表示を音声で指示すると、確認の上、光ディスク装置のカラーディスプレイ上に検索結果のカラー写真が表示される。

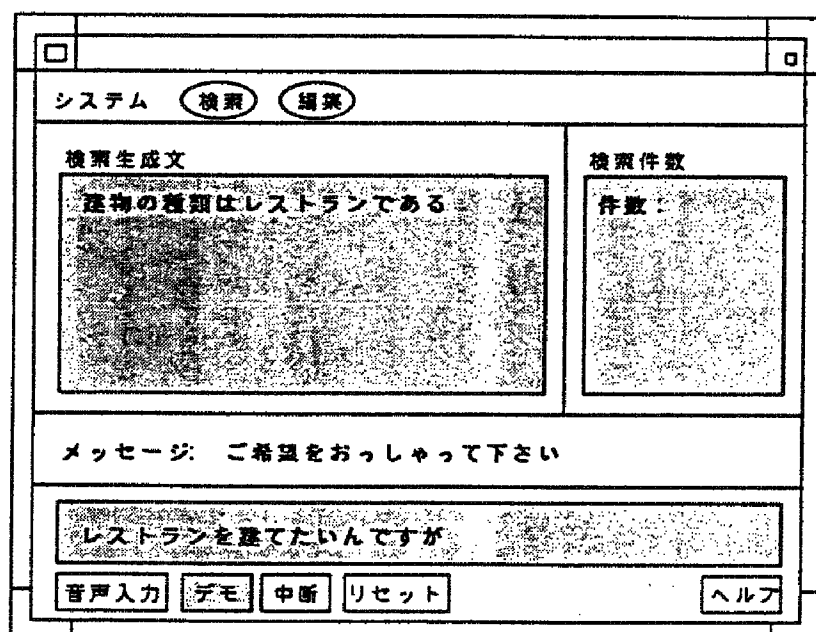


Fig. 7

音声による検索画面例

8. 試行結果

本システムは展示会場の騒音下で、音声研究とは全く関係のない社外の案内嬢2名によりデモンストラーションが行われた。

発声文章の形式は特に細かい指示を与えなかったため、口語文生成ルールで生成される文章以外の文章の発声も数多くなされ、一致する文章がないため非注目語の部分では誤認識が生ずるケースも多々在るが、検索に重要な注目語（キーワード）の誤りはほとんど生じず、検索はスムーズに行われた。口語文生成ルールで生成される文章の種類がかなりバラエティに富んでおり、一致する文章が無くとも、類似した文章があるため非注目語（キーワード）の認識誤りの注目語への影響を小さく押さえる事が出来ているものと思われる。このことは、タスクは限定しても出来るだけ発声には制約を設けない状況で高い理解率を得たい、という期待の実現可能性を示していると言えよう。

9. おわりに

本研究では、音声対話の良さを保ちながら現実的アプローチで音声対話システムを実現する可能性を検討し、ある程度の見通しを得ることができた。

しかし、音声対話研究には本質的に難しい数多くの技術課題が残されている。また、ここで示したよう

な現実的アプローチによってある程度の範囲での実現の可能性が示されたとしても、開発した対話システムをどの様に評価したら良いか、という問題も有る。ユーザの主観的満足度（例えばMOSによる評価）や目的の完了率などが考えられるが、その結果が否定的評価であった場合、システムを構成している個々の技術の何処を改良したら良いか、という指針とは直接的に結び付かない。今後の大きな課題の一つである。

また、その技術がどのようなアプリケーションに応用できるか、という事を考えると、適用可能な範囲を明確にするために、対話を用いるアプリケーションを技術的に分類整理する必要も有る。個々のアプリケーションと対話技術を結び付けるアプリケーション記述手法等の周辺技術の開発も重要である。

認識処理の実時間化を担当した村山秀樹君と生成文法の作成および音声データベースの作成を担当した井澤真理子君に感謝する。

参考文献

- [Amano1989] Amano et al., "On the Use of Neural Networks and Fuzzy Logics in Speech Recognition", Proc. of IJCNN 89, I-301
- [Hataoka1989] Hataoka et al., "Large Vocabulary Speech Recognition Using Concept Networks", Proc. of IJCNN 89, II-576
- [Komatsu1986] Komatsu et al., "Prosodic Aids to Structural Analysis of Conversational Speech", Proc. of ICASSP 86, 33.10, pp2283-2286
- [Komatsu1990] Komatsu et al., "Conversational Speech Understanding Based on Cooperative Problem Solving", Proc. of ICSLP, 27.10, pp1225-1228
- [Komatsu1991] 小松昭男, " 会話音声理解によるマン・マシンインタフェースに関する研究", 学位論文
- [北原1991] 北原他, " 音声入力による情報検索システムにおける口語文受理文法の検討" 音学講論 3-5-7, 平成3年3月
- [村山1991] 村山他, " 連続音声入力による情報検索システム" 音学講論 3-6-11, 平成3年3月

自然な模擬対話を収録するために

菊池英明 小林哲則 白井克彦

(早稲田大学・理工学部)

1. はじめに

音声対話システムに要求される基本機能を検討する際の材料として必要とされる模擬対話の収録において、いかなる条件を設定することが自然で有用な対話の収集につながるのかという問題について、我々が実際の模擬対話の収録を通じて得た経験的知識を紹介する。

音声対話の研究は、技術的に多くの課題を含んでいることが予想され、また、研究の歴史が浅いこともあって、現状では対話現象の一面に注目して研究を進めざるをえない状況にある。しかも、このとき音声対話にはどのような問題点が存在するのかすら現状で明らかとはいえず、従って、どのような問題に焦点を当てて研究をすることが実質上価値あることなのかについても当然不明である。これらの問題は、タスクに大きく依存することから、タスクの性格の分類と関連させて検討を進めることが望まれている。このためには、模擬対話の収録とその分析が重要な役割を演じることが予想される。そこで、我々は車のナビゲーションをタスクとして模擬対話の収録を開始した。しかし、自然で（ここで、「自然」というのは、ユーザの立場で対話に参加した者が、違和感なく対話を続けることができることを指すものとする）かつ研究対象として魅力ある対話を模擬対話の収録を通じて得ることは思いのほか困難なものであった。これまでに、5回の条件変更を重ねて、試行錯誤を続けている。いまだ十分な条件設定が確立されたとはいえないが、本報告では、これまでの経緯をまとめてみることにより、自然で有用な対話の収集を行うための条件設定について考えてみたい。

なお、ナビゲーションをタスクとして選んだのは、ドライバーの置かれた状況から、インタフェースのメディアとして音声に期待されるものが大きいと判断したためである。

2 対話データ収録条件の変更の過程

車中で、音声入出力が重要な役割を果たすことが期待される機能として以下のものが挙げられる。

(1) 運転中の逐次的な道案内（指示）

(2)目的地までの基本ルートの生成

(3)目的地, ルートの相談

(4)カーオーディオ, 自動車電話に代表される車内付属機器の取扱い

ここで, ルートの生成というのはユーザが確固とした目的地を頭の中に描いており, ルートの選択条件も比較的明快な条件の下で, システムがそれを聞き出し, ルートの案を提示することを意味し, 目的地, ルートの相談とはユーザがそれらにはっきりとは決めていない条件下で, システムがコンサルテーションを通じて目的地, ルートを決めようとすることを意味する(例えば, 「ドライブに行きたいのだが, いいところはないか」といった類の意識のユーザにドライブコースを提示するような状況を指す). 以上に挙げたもののうち, (4)に関しては, 単純なコマンド入力に近いものであるため, 対話的要素は低いことが予想される. ここではの対話の現象を調べることに興味を持ったため, さしあたり(1)(2)(3)の機能について模擬会話を収録し, 対話現象の研究にふさわしい問題設定を模索することにした.

2. 1 何も条件をつけないで模擬対話を行った場合

—興味ある対話の現象は, 設定したタスク内において
どのような機能の実現時に現れるか—

まず, 実際に自動車内で道案内のやりとりを人間が行い, 対話としてどのようなバリエーションがあるかを調べようとした. 現在地と目的地を設定した以外は何も条件を付けなかった. あくまでも人間同士のやりとりを収録することが目的で, 特にナビゲータ役が機械であることは意識させなかった.

収録された対話を書き起こしたテキストの一例を例1に示す.

対話を見たところ, 逐次的な道案内(ここでは, 「次の角を右に曲がって下さい」の類の案内をこう呼ぶことにする)に関しては, 視覚的情報が絡むことによって生ずる対話の複雑性がその分析を困難にしていると思われた. 例えば, 「ここを左折ですか?」という発話は視覚情報のない設定ではありえない. また, 走っているとき, 運転者が不安になっているのを察してナビゲーターが「当分まっすぐ行って西落合一丁目の交差点で左に曲がります.」と発話して確認を行うのも同様である. こういったシステムとユーザが視覚情報を共有することに基づく発話は逐次的な道案内では頻繁に現れ, ナビゲーションタスクでは重要な側面であることが予想される. しかし, 現状でこのような視覚情報を扱うことは困難であり, このような条件設定で実験を進めることには無理がある. また, 適切に指示がナビゲータから出される限りにおいて, ユーザの応答が単調なものになってしまうのも対話の現象をみる観点からは好ましくないように思われた. そこで, 以後の実験では, この逐次的道案内の機能は対象からはずすこととした.

収録した対話の中には, 必ず冒頭に目的地までの基本ルートを生成するための対話(上の(2))がある. これは, 対話開始前に, 設定として目的地を決めただけ

であり、そこまでのルートは運転者もナビゲータも決定していないからである。この部分では地図程度の視覚情報だけで対話を行えるという点で、我々の目的に適していると思われる。また、上で挙げた機能のうち、(3)の目的地、ルートの相談に関しては、その現象が設定したタスク内での対話中に頻繁に現れ、対話の現象を扱うには適しているという点では(2)と同様であるが、(2)と比較して対話中で話者が扱う知識がより多いことが予想される。これらの理由から、次の段階では、この目的地までのルート生成機能のみに注目して対話収録を行うことにした。話題が発散し、観測される対話の現象が密度の薄いものになる危険性があることから、大まかなシナリオをシステム側に用意することにした。

2. 2 ルート生成機能において、大まかなシナリオを システム側に設定した模擬対話

対話中では目的地までのルート生成しか行わないという条件設定で、収録を行った。この条件下では実際に自動車に乗り込んで対話を収録する必要性はないものと考え、実験は室内で行うことにした。

ユーザは、横浜ベイブリッジまでのドライブと東京ディズニーランドでの待ち合わせのどちらかを望んでいるものと仮定し、システム側では、それぞれについてユーザから聞き出すべき情報を数点設定した。

この条件における収録対話の例を例2に示す。

結果として以下の問題点が浮かび上がった。

①条件設定の善悪：

システム側での対話方針（質問事項）を予め設定したため、設定された条件に思考が拘束され、型にはまった対話が行われてしまい、かなり不自然である。

②システムの役割の徹底、システム役の条件：

システム役をした人の道に関する知識が不完全であるため、システムがユーザに相談する場面が多かった。この現象は分析する際、複雑度を増すので、好ましくない。また、実際システムの知識は完全であることを期待できるから、この意味でも、ここで得られた対話は実現しようとするシステムとの間で予想される対話と隔たりがある。人対人の対話をそのまま扱うのは難しすぎ、また、システムを制作する観点からは実際的ではないので、人対機械に近いデータを見たいという要求が生じた。このためには、ナビゲータの機能を整理し、その役につく人がシステムの役割を完全に遂行できることが重要と考える。

また、模擬対話にユーザ役で参加した人から、ルートの詰め方が不自然であるとの指摘を受けた。システム役は、ルート生成の方法として、先に一方的に1つのルートを提示してそれに関してユーザの意見を尋ねる方法と、部分的に選択の可能性を示しながらユーザとの話し合いで協同生成していく方法とを、ユーザに応じて使い分けたが、それらの方法にシステム役の一方的な思いこみがあり、まだ理想的なシステムの役を十分演じきれていないことの現れと考えられる。この

辺は、頭で考えて役を演じることの難しさの一面を示している。

③ ユーザ役の条件：

実際にはユーザの役割をした人は、明らかに自分が知っていることを尋ねている。横浜ベイブリッジまでの道はほとんどの人が知っており、それをあえて聞くと設定したためである。知っていることを尋ねるよう指示された場合、視点に新鮮味が欠け、対話を意外性の少ない型にはまったものになっている原因となっている。ユーザはこれから聞こうとする事について本当に知らなくて、かつその事項に興味を持っていることが望まれる。

また、ユーザにとってみれば、単に目的地を設定されただけであるため、そこへ行く目的意識がなく、発想が貧困になっている。確固たる状況設定をユーザ自身が持って、模擬対話に望む必要がある。

④ 対話の主導権：

対話中の主導権の所在について、模擬対話を意識しすぎたためにシステム主導になってしまった例が多い。自然対話においては、話題の制約がないこともあって、主導権の所在は流動的である。ルート生成のような目的を達成するための対話では主導権は動きにくいことも予想されるが、いずれにせよ、対話の流れはも、どちらに主導権があるかに依存する。従ってシステム主導とユーザ主導で対話を収録してみて、主導権をあらかじめ決定するとどのような流れになるかを調べる必要がある。

2. 3 主導権を予め設定した対話

前節④の問題に着目し、システム主導型の対話を3、ユーザ主導型の対話を3、横浜ベイブリッジまでのドライブというタスクで収録した。あらかじめドライブ（遊び）で横浜方面に行くという目的をもたせた。システム側はユーザ主導型ではユーザの要求の最低限の情報を伝えるようにし、システム主導型ではシステムから積極的にルート生成を達成するためにユーザの要求を尋ねるようにした。

一例を例3に示す。

その結果、システム主導型の対話は今すぐにでもシステムとして実現できそうな程、対話的な問題点がない。シナリオを設定したとすれば模擬対話ではそのシナリオからそれることはない。また、ユーザ主導においては、実際場面で起こりそうな意外性がなく、あらかじめ予測できそうな発話しかない。このように主導権を収録前に決定すると、話題の広がりがなくなり、ますます自然性がなくなるようである。結論として、主導権の移動は意識的にも無意識的にも不自由であってはならないといえる。

2. 4 ユーザに対する状況説明の徹底

ユーザ役に自分のおかれた状況設定を徹底して説明し、それを十分理解させた上で模擬対話の収録を行なった。その際、主導権の所在は問わないことにした。

設定する状況は3つ用意した。

結果を例4に示す。

比較的対話らしいデータが収録できたと考えている。しかし、予め与えた状況設定に基づいて話した発話はやはり偽物で、生きた設定、つまり日常での対話とは異質である。設定されるべき状況のバリエーションは数多くあることが予想され、その真実味も室内で頭で考えて設定しただけでは限界がある。模擬対話を行う際の生きた状況設定を行い、対話の自然性を回復するには、再び外にでて実環境に身をおいてデータを収録することが必要であることを感じた。

2. 5 フィールドにおける対話データ収集

自動車内で、ユーザ役に状況設定をひとつとって認識させただけでそれ以外には何も条件を付けずに対話データの収録を行った。つまり、2. 1の収録条件と場所と対象とする対話の機能を絞っていること以外は全て同じである。

結果を例5に示す。

以前の収録対話より長い対話が得られた。これは、自動車内で収録することにより、実際の場面に近づいたため発想に柔軟性がでたためと思われる。このことから、模擬対話を収録する際には、より実際の場面に近づいた状況で収録するべきであると考えた。

3. むすび

以上、5回の模擬対話収録実験を通じて我々が得た経験を述べた。現状での結論は以下のように要約される。

①対話の主導権の移動は意識的にも無意識的にも不自由であってはならない。

このためには、システム側での型にはまった対話方針（質問事項）の設定は好ましくない。

②システムの機能の整理、役割の徹底は重要である。

③システム役の人は、役割を完全に遂行できることが望まれる。対象分野に対しかなりエキスパートであるべき。（知識があることを装うのは難しい）

④ユーザ役は、聞こうとする事について本当に知らなくて、かつその事項に興味を持っていることが望まれる。（知識が無いことを装うことも難しい）

⑤ユーザは、確固たる状況設定を持って模擬対話に望む必要がある。

⑥より実際の場面に近づいた環境で収録するべきである。

得られた結論は、最初からそうすることが望ましいことが分かっているが、種々の理由（例えば、⑥の例では、事故を恐れるが故、できるだけ学生にはハンドルを持たせないで済ませたかった）で避けていた側面もある。しかし、それぞれの要因の重要性を実感できた意味において、試行錯誤もまた意味深いものであったと考えている。

今後、収録の際の制約を明確に規定すべきであり、制約を少しづつ変化させて収録するという方向を考えている。システム側の発話により対話全体がどのように制御されるかについても、システムの発話方針を意識的に変化させてみてその後の対話内容にどのような変化が見られるかを細かく見ていきたい。また、模擬対話に限らず、日常での対話を数多く収録することも必要と考えている。実験を意識すると、どうしても身構えてしまう。発話内容は単純なものに片寄る可能性もあるが、さまざまな対話の現象が観測できることが期待できる。

ここでは、決まったタスク、それもタスク内でシステムのある特定の機能を利用しているという条件下で、自然な模擬対話を収録する条件について検討した。しかし、種々の異なるタイプの対話についてその性格を分析するとともに、この性格と模擬対話の条件設定に要求される要件、あるいはそこで観測される重要な対話の現象を系統づけて整理することが、同種の実験を行っている様々な機関に個々に積み上がっているノウハウを統合する意味で非常に重要な仕事と考えている。

<対話例1>

現在地：早稲田大学理工学部

目的地：哲学堂

U：哲学堂にいきたいんだけど。

S：まずこれを右に曲がって明治通りにでましょう。

S：明治通りを左に曲がりましょう。

<左折>

S：明治通りにでたら、新目白通りを左に曲がります。

<左にうつる。左折しようとする。>

S：ここは違う。

S：新目白通りだよ。

U：新目白通りの場所は？

S：2個目の交差点です。

U：この次の信号ですね。

S：そうです。

U：交差点の名前は？

S：高田橋です。

U：ここを左折ですか？

S：そうです。

<左折>

S：とうぶんまっすぐ行って、西落合一丁目の交差点で左に曲がります。

U：時間はどのくらいかかる？

S：10分くらいです。

U：距離は？

S：3kmくらいです。

U：もう一度、次に曲がるのはどちらですか？

S：左です。

S：これを左です。

<左折>

U：しばらくまっすぐ？

S：少しいくと、左に哲学堂の入口があります。

<対話例2>

ルート生成対話（模擬対話）

現在地：早稲田大学理工学部西門

目的地：東京ディズニーランド

現在時刻：平日4時30分

U：今、早稲田の理工の西門にいるんだけど、東京ディズニーランドに6時

までに行くにはどういう道でいったらいいと思います？。

なるべく、大きな道がいいんですけど。

S：時間の余裕がないので、首都高を利用するのがベストですが、金銭的にはどうでしょうか？

U：金銭的にはいいんですけど、本当に込んでないんですか？

S：一般道も高速道路もこんでいますが、首都高速の方が確実に流れると思います。

U：では、首都高はどこから乗ればいいと思いますか？

S：ルートとして分かりやすいのは新宿のランプに乗る方法です。

距離的には飯田橋から乗る方法がいいと思われませんが、どちらにしますか？

U：では、新宿から行きましょう。

何か目印はありますか？

S：はい。

理工学部の前を通っている明治通りを南へ行きます。

そうすると、新宿で甲州街道に出ます。

そして甲州街道を右に曲がると、首都高に出る案内が出ています。

その案内に従えば、新宿ランプから首都高にのれます。

U：では、それで行ってみましょう。

S：それでは、新宿のランプから乗って、首都高4号線を環状線の方面へ向かい
ます。

そして首都高速環状線に入って江戸橋のランプから箱崎に行くと湾岸方面へ行く案内が出ます。

そこから、湾岸線に入って、ディズニーランドへ行けます。

ディズニーランドへ降りるには浦安のランプで降ります。

<対話例3>

ルート生成対話（模擬対話）

現在地：高田馬場

目的地：横浜

1991, 10, 8

S：目的地はどこですか？

U：横浜です。

S：特に時間の指定はありますか？

U：特にありません。

S：では、ゆっくり行ってもいいわけですね。

U：かまいません。

S：それでは、寄りたいところがありますか？

U：特にありません。

S：わかりました。

それでは、ルートを考えます。

まず、大きく分けて、高田馬場から一般道を通して第3京浜を使う方法と、首都高速を使う方法の2
つがありますが、どちらにしますか？

U：時間の早いのはどちらですか？

S：時間は、第3京浜を使う方が早いと思います。

U：では、第3京浜を使いたい。

S：わかりました。

第3京浜の入口の玉川インターに行くには、山手通りと環7と環8があります。

混雑度はどれも同じだと思います。

わかりやすさは環8が直接第3京浜に乗れるので、分かりやすいと思います。

どれにしますか？

U：そうですね。環8からのるんでしょうね。

S：それでは、環8を使う方法でよろしいですね？

U：はい。

S：それでは、環8に出るまでに、早稲田通りと甲州街道と246を使う方法があります。混雑度に関
しては、早稲田通り、246、甲州街道という順でしょう。

どれにしますか？

U：甲州街道にしたいと思います。

S：わかりました。

高田馬場から明治通りを南に行って、新宿3丁目で甲州街道を右折して、環8を左折して、第3京浜
に乗って横浜に行くということよろしいですか？

U：はい。

<対話例4>

ルート生成対話（模擬対話）

現在地：高田馬場

目的地：外人墓地→山下公園

寄る所：ベイブリッジ

トピック：目的地変更

U：横浜の外人墓地にいきたいんだけどどういったらいいでしょう。

S：時間の指定はありますか？

U：特にありません。

S：わかりました。どこか寄りたい所はありますか？

U：ベイブリッジに寄ってから行きたいと思いますが。

S：それでは、現在地高田馬場から横浜ベイブリッジを通して横浜の外人墓地に向かうルートを生成します。

ベイブリッジに行くまでに、主に、湾岸を通る首都高速を使う方法と、一般道から第3京浜を使う方法がありますが、どちらにしますか？

U：時間的にはどちらがはやいですか？

S：時間的には一般道を通って行った方が早いと思います。

U：じゃあ、下から首都高速を使わない方でいきましょう。

S：わかりました。

それでは、高田馬場から早稲田通りを西にいて、環8との交差点で左折して下さい。

環8を南に下ると、246号を過ぎたところで、第3京浜の入り口の玉川インターチェンジがありますので、そこから第3京浜に入して下さい。

途中で、首都高速に合流して、大黒埠頭方面に向かって下さい。

生麦ランプで羽田方面と大黒埠頭方面に分かれますが、大黒埠頭方面に向かって下さい。

途中で横浜ベイブリッジを通ります。

そのまま、まっすぐ山下町方面に向かい、新山下ランプで首都高速を降りて下さい。

一般道を横浜公園方面に向かい、谷戸橋という交差点を左折すると外人墓地に行きます。

ただし、外人墓地付近には駐車場がないのですが、よろしいですか？

U：外人墓地には駐車場がない？

S：はい、付近に駐車場はありません。

U：そうすると、山下公園。

山下公園には駐車場ありますか？

S：はい、山下公園には駐車場があります。

U：それじゃあ、そこに止めたいと思いますが、駐車料金はいくらぐらいでしょうか？

S：1000円です。

U：はい。わかりました。

S：それでは、最終目的地は山下公園でよろしいですか？

U：はい。

S：それでは、新山下ランプで首都高速を降りるところまでは同じで、そこから、山下公園方面に向かいます。

すると、山下橋という交差点を右折したところに山下公園の駐車場の入り口があります。

<対話例5>

ルート生成対話

1991.12.24 PM9:00

現在地 早稲田大学理工学部前

目的地 横浜ベイブリッジ

U：横浜のベイブリッジに行きたいんですけど、どういったらよろしいんでしょうか？

S：うーんと、ドライブですか？

U：あ、はい。

S：ドライブ。じゃあ、えーと、ベイブリッジを見るというのが目的だね。

U：そうですね。

S：うーん、じゃあ、時間は何時でもいいということで。

U: そうですね、ただ、できるだけ、すいている方がいいんです。

S: あ、はい。それじゃあ、まず、ベイブリッジっていうのは、首都高速の上にあるんですけど、

U: はあ。

S: まあ、どこからか首都高速に乗らなければいけないんだけど。
ええ、高速を使いたいですか、それともなるべく使わないようにしたいですか？

U: うーん、なるべく使わない方がいいです。

S: そうか、じゃあ、そうすると、今は、現在地は高田馬場だから、

U: あ、高速っていうと、

S: うん。

U: 首都高速ですか？

S: そうですね。首都高速です。

U: じゃあ、首都高速の結局どこから乗らなければいけないということですか？

S: そうです。

U: それだったら、結局首都高は料金均一なんで、別に多く乗ってもかまわないですけど。

S: あ、そうですね。わかりました。まあ、この時間だったら、首都高速はそんなに渋滞はないと思うんですけども、

U: ええ。

S: 一部、渋滞があるところがあると思うんで、

U: ああ。

S: 下の方を通ったほうが早い場合もありますが。

U: では、その込みそうな場所を避けて、首都高に乗るルートで。

S: あ、はい。わかりました。それだったら、ええとね、ここから、渋谷とか新宿とか羽田とか、そっちの方は込むと思うんで、まず、一般道を通して、西に行って。
まあ西に向かう通りは、甲州街道とか、246とか、目黒通りとか、早稲田通りとかいろいろあるけども、環8通りにぶつかったら、環8通り沿いに第3京浜の入り口があって、この第3京浜をまっすぐ行くと途中で、首都高速に乗ることができるんですよ。

U: ああ。

S: で、そうすると、わりと、首都高速の込まないところに着いて、すぐ、ベイブリッジに着くんだけれども。
そんな感じで、いい？

U: そうですね。ただ、環8に出るまで、どういうルートにしたらよろしいでしょう？

S: そうですね、それは、混雑度を考えると、あと、道の大きさなんかも考えると、20号あるいは、246とあまり変わらないと思うけど。

U: ああ。ただ、246は三軒茶屋のあたりまで行くのが、普段だったら込むとおもうけどな。

S: そうですね。

U: そこの所は、大丈夫でしょうか、この時間帯は。

S: そこは込みますね。だから、えっと、ここからだったら、早稲田通りで、まず、環7ぐらいまで出て、で、そこから、大きい通り、20号とかに出るとか、246に出るっていう方法ならば、渋谷とかのあの辺りまで避けられると思うけど。

U: ああ、そうですか。

S: うん。

U: それじゃあ、まず、いったん環7にでて、そこから246に出るというルートで。

S: それで、いいですか？

U: はい。あと、環7まではどのルートでいったらよろしいでしょう。

S: そうだね、ここからだ、早稲田通りに戻って行くか、あるいは大久保通りで行くか、あるいは新宿まででて、20号で行くか、まあ、道の大きさは随分違うけど、混雑度はそんなに変わらないと思う。

U: そうですね、ただ、新宿は人が多いので、できれば避けたいんですけど。

S: そうですか。じゃあ、早稲田通りで行きましようか。

U: はい。じゃあ、早稲田通りを使って環7まで出て、そこから、246で環8に出て、第3京浜を使うという感じで。

S: それで、いいですか？

U: はい。

S: じゃあ、そうしましょう。

ATR 対話コーパスを利用した音声認識のための構文規則

保坂順子、竹沢寿幸、江原暉将¹

ATR 自動翻訳電話研究所

あらまし 音声認識では、構文情報が制約として有効に働くことが期待されている。しかし、日本語のように語順が比較的自由な言語では、制約として有効に働く規則を構築することが難しい。我々は、理想的な構文規則の構築を目指し、対話コーパスを積極的に活用することを提案する。本稿では、はじめに、制約の緩い規則により、音声認識傾向を調べる。さらに、その考察に基づき、規則を改良する。その際、実際の言語運用を反映させるため、どのように対話コーパスを利用しているかを示す。最後に、このようにして構築した構文規則を取り入れることの効果を、実験により確認する。

1 はじめに

話しことばの音声自動翻訳の実現を目指して [1]、音声認識に有効に働く構文規則の構築を検討している。現在、国際会議に関する問合せをドメインとしている。

日本語の音声認識は、孤立発声単語の処理から文単位連続音声の処理へ研究対象が発展しつつある。この中間段階として、我々は、文節単位で発声された文連続音声の処理を試みている。音声認識には、構文情報を利用した HMM-LR 方式 [2] を適用している。HMM-LR 方式では、文脈自由文法の枠組で記述された構文規則を使い、認識の最小単位である音韻の予測を行ない、同時に HMM (Hidden Markov Model) 音韻モデルと照合するということを実現している。

この HMM-LR 方式により、語順が定まっている文節内については、高い認識率が確認されている [2]。一方、文節間には、係り受け関係を導入して文認識を行なうことが提案されている [3]。これは、文節を基に文を組み立てる規則を文節内と同様に文脈自由文法の枠組で一般的に記述するのが困難なためである。日本語は文節の出現順序が比較的自由であり [4]、かつ文節の省略も頻繁に行なわれる。我々が対象としている話し言葉では、特にその傾向が強く、文脈自由文法の枠組では記述しきれない。係り受け情報が有効に働くことは、少数語彙の処理において確認されている [5]。しかし、大語彙を扱う場合、音声認識に制約として働くように、一貫性を保った係り受け関係を適切に作ることは難しい。そこで、文節を基に文を組み立てる規則も、文脈自由文法の枠組で扱う可能性を調べる。

自然言語処理では格に注目して文の規則を構築することがある。構文規則を制約としてとらえた場合、文節の出現順序や省略を考慮してすべての規則を列挙すると、それは、これらの制約を与えないのとほとんど等価となる。さらに、話しことばは、書きことばに比べて

¹現在NHK

言い回しが自由である。この自由度を反映させた構文規則を音声認識に適用すると、正しいものが認識されにくくなる。

音声認識において、構文情報は次の音韻の予測および制約として利用され、容認性の低い音韻列を排除するために使われるのであり、認識された文の構造がどのように解析されるかは、現段階では自然言語処理の構文解析ほどの重要性を持たない。従って、タスクに合わない文節または文の生成は極力避けるのが望ましい。さらに、音声認識の傾向を考慮することも望まれる。しかし、これは、文法作成者の直感だけに頼っていたのでは、非常に達成し難い。そこで、我々は、対話コーパス [6] を利用して実際の言語運用を調べ、それに基づき理想的な構文規則の構築を目指す。対話コーパスには、書き言葉に近いキーボード会話と、話し言葉である電話会話が収録されており、文字、形態素、係り受けなどの情報を利用した検索が可能である [7]。従来、構文規則は書き言葉を中心に扱ってきたが、両者を比較することにより、話し言葉との違いが明確になることが期待される。

本稿では、まず HMM-LR 音声認識手法の認識傾向を考察する。次に、制約の緩い構文情報を使い音声認識において誤認識の対象となりやすい要素を明らかにする。さらに、その考察に基づき、制約として働きそうな構文規則を作る。その際、実際の言語運用を反映させるため、どのように対話コーパスを利用しているかを示す。最後に、このようにして構築した構文規則を取り入れることの効果を、実験により確認する。

2 音声認識傾向

音声認識には、文脈自由文法の枠組で定義した規則により次の音韻を予測する、という手法を実現した HMM-LR [2] を利用している。この手法を適用した場合、どのような制約を強めるべきか調べる [8]。そのため、単語の後にはどの単語が来てもよいという、すべての単語の連鎖を許す規則を定義した。この規則を文節単位の音声認識²に適用した場合の認識結果例を (1), (2) に示す^{3 4 5}：

(1) |sochirawa|

```
> 1: sochira-wa
   2: sochira-wa-hu
   3: sochira-hu-wa
   4: sochira-hu-wa-hu
   5: sochira-wa-hu-hu
```

² この実験の設定はグローバルビーム幅 16、一本の枝からのローカルビーム幅 10 である。

³ “|” で囲まれたものは、入力文節である。

⁴ “>” は、正しく認識されたものを示す。

⁵ “ng” は、鼻音化したガ行音、“N” は撥音を表す。

(2) |arigatougozaimasu|

1: ari-nga-to-wa-eN-hu-su-su-su

2: ari-nga-to-wa-eN-hu-su-su

3: ari-nga-to-wa-eN-hu-su-su-u

4: ari-nga-to-wa-eN-su-su

5: ari-nga-to-wa-eN-hu-su-su-su-u

HMM-LR では、左から右への探索を行っているので、文節のはじめが誤って認識されると、文節の正しい認識は望めない。しかし、(1) からわかるように、文節のはじめが正しく認識されても、文節末の認識は、不安定である。(2) は、[tou] が、[to] と誤って認識され、それ以降正しい音韻列が見つけられなかった例である。(1) (2) 共に、認識を誤った時点から、音韻数の少ない単語⁶を認識候補に選択している。

これを回避するためには、音韻数の少ない単語について制約を強めることが考えられる。音韻数の少ない単語としては、助詞、数詞、副詞の一部などがある。

3 音声認識における誤認識傾向

文の構文情報を予測として音声認識に適用した場合に、誤認識されやすいものの傾向を調べる。我々は、文節単位の発声を入力としている。そこで、HMM-LR 方式による文節の音声認識⁷から得た候補を、文の構文情報を使い絞り込むという二段方式を採用した [9]。

文節の音声認識実験は、137 文を構成する 353 文節について行なった。その結果、1 位で 68.3%、5 位までの累積で 95.5% の認識率だった。

文の構文情報を音声認識に適用する際に、誤認識されやすい要素を調べる実験は、文節音声認識の結果得られた文節候補 5 位までを基に行なった。実験のために、許容率 [9] の高い文の規則を用意した。この規則の中で、文として考慮しているのは、感動詞からなるもの、接続詞で始まり述語文節で終るもの、接続詞を伴わず述語文節で終るものの 3 種類である。終端記号としては、前もって構築した文節内の規則で使っている文節カテゴリーを利用した。文節カテゴリーとしては、名詞文節、副詞文節、連体修飾文節、接続助詞で終る述語文節、助動詞で終る述語文節、引用を扱う文節など約 25 種類がある。

この規則は、文節単位の音声認識で 5 位までの候補になった文節のすべての組合せを許したもののうち約 80% のものを文として容認した。実験の結果、文の候補になったもので、発声された入力との文字列のマッチングがとれていない例を以下に示す⁸：

1: 会議に申し込みたいのですが。

1a: kaingi-ni moushiko-mi-tai-N-desu-nga

1b: kaingi-ni moushiko-mi-tai-no-desu-ka

⁶動詞、形容詞など、活用のあるものは、活用したものを一単語としている

⁷この実験の設定は、グローバルビーム幅 100、一本の枝からのローカルビーム幅 10 である。

⁸“N” は、撥音を表す。

2: こちらは会議事務局です。

2a: kata-wa kaingizimukyoku-desu

3: どうもありがとうございました。

3a: go-o aringat-ou-gozaima-shi-ta

3b: go-mo aringat-ou-gozaima-shi-ta

3c: mono aringat-ou-gozaima-shi-ta

4: 名前は清水太郎です。

4a: namae-e-wa shimizu-taroo-desu

4b: nana-e-wa shimizu-taroo-desu-shi

5: ご住所とお名前をお願いします。

5a: gozyuusho-to onamae-o onengai-shi-masu-shi

準体助詞の「の」は、文候補 1a のように「ん」と認識されやすい。これらは互換性があり、そのニュアンスの違いを構文規則で表すのは難しい。文候補 1b は、疑問文として認識されている。文単位の構文規則では、疑問文、平叙文または命令文などを区別することはできない。文候補 1a-b は、文の構文規則で扱える範囲を越えている。

文候補 2-5 において、形式名詞、数詞、格助詞「へ」および接続助詞の 4 種類の要素を含む文節が文中で容認性が低いのが目立った。

- 形式名詞を含む文節

2a は、「緑の制服を来ているかたは会議事務局です。」とすれば容認性が高くなるであろう。また 3c は、「高価なものありがとうございました。」とすれば普通の文になる。

- 数詞を含む文節

3a と 3b では、数字の 5 が使われている。これらの文は一見非文に見えるが、トランプをしている状況を考えると、普通の文である。5 が出れば、自分は上がれるという時、誰かが 5 を出してくれれば、「5 をありがとうございました。」と言うであろう。また、あと 3 と 5 が必要だという時、誰かが 3 を出してくれ、さらに続いて他の誰かが 5 を出してくれれば「5 もありがとうございました。」と言える。

- 格助詞「へ」を含む文節

4a と 4b は、方向性を表す格助詞「へ」と方向性のない「です」が共起しているため、非文である。「名前へは」と「7 へは」は非文節にも見えるが、「住所へは結構ですが、名前へは振りがなをお願いします。」または、「1 から 6 にはコメントを書きましたが、7 へはまだ何も意見を書いていません。」などは容認性が高い。

- 接続助詞終止の文末文節

4b や 5a では文末に接続助詞「し」が使われている。これらは、「清水太郎ですが。」「ご住所とお名前をお願いしますので。」ほど自然ではない。文末に使われる接続助詞は、「が」や「ので」など一部のものに限られるであろう。

構文情報を音声認識に効果的に利用するには、これらの形態素を含むものに、特に強い制約を加えることが考えられる。2節では、HMM-LR 方式を使った場合、音韻数の少ない単語が認識候補になりやすいことがわかった。誤認識の対象になりやすい、形式名詞、数詞、格助詞、接続助詞などには、音韻数の少ないものが多く、この傾向を反映している。4節、5節では、制約を強めるために、どのように対話コーパスを活用しているか、形式名詞および接続助詞を例に紹介する。

4 形式名詞

「こと」や「もの」など、名詞の一種である形式名詞について考察する [10]。形式名詞は、実質的な意味がうすく、連体修飾を受けてのみ用いられる [11]、と言われる。これらは、確かに、例文 6 や 7 のように連体修飾されることが多い。

6: 昨日言ったことは取り消したい。

7: 値段が高いものは必ずしも質がいいわけではない。

一方、例文 8 や 9 のように、修飾を受けないこともある。

8: それは、ことですね。

9: ものは確かだ。

そこで、「こと」や「もの」には、二種類あると定義してみる。一つは例文 6 や 7 のように使われる形式名詞であり、もう一つは例文 8 や 9 のように使われる普通名詞である。しかし、これは、文の解析には役に立つが、制約という観点からは、意味のない区別になる。

「こと」、「もの」さらに「方(ほう、かた)」が、実際にはどのように連体修飾を受けて使われているかを調べるため、対話コーパスを検索した [6, 7]。ここでは、最も多いと思われる活用語の連体修飾を扱う。検索対象は、我々が音声自動翻訳で扱っているのと同じ国際会議に関する問い合わせをドメインとしている対話である。書き言葉に近いキーボード会話は、3318 文 (113 会話) であり、話し言葉である電話会話は、7351 文 (96 会話) である。

表 1: 形式名詞 - こと, もの, 方 -

	キーボード会話		電話会話	
	出現頻度	%	出現頻度	%
活用語の修飾あり	358	63	774	40
活用語の修飾なし	214	37	1145	60
文頭	0	0	0	0
合計	572	100	1919	100

表 1 から、活用語の連体修飾を受けることを形式名詞の特徴にすると、書き言葉における被覆率は 63 % になるが、我々が対象としている話し言葉においては 40 % にしかならないことがわかる。さらに、活用語に連体修飾されないものについて調べたところ、ほとんど

のものが、名詞に助詞の「の」が後続したものによって修飾されていることがわかった。また、動詞に助詞の「と」が後続し、さらに「の」が後続したものに修飾されるもの、「どんな」や「その」などの連体詞に修飾されるものなどが若干あった。何の修飾も受けないものは、キーボード会話に次の一例だけあった：

恐らく教授にことの次第を伝えることが良いと思います。

対話コーパスでは、「こと」、「もの」と「方（ほう、かた）」を含む文節が 2491 例あったが、そのうち修飾されないものは一例だけだった。これから、「こと」、「もの」と「方」は形式名詞であり、形式名詞は、必ず修飾されるという制約を加えても、その被覆率は 99 % を越えることがわかる。しかも、我々の扱っているコーパスで修飾されずに使われるものは「ことの次第」のような固定化した言い回しであるので、それらを文脈自由文法の枠組で扱うことはできる。

5 接続助詞

「し」「が」「ので」などの接続助詞が、文末で使われることについて考察する [12]。接続助詞は、二つの文の意味関係を示し、これらを一つの文にまとめあげる働きを持っている。そのため、文中で使われることが多い。しかし、「会議に申し込みたいのですが。」のように、文末で使い、余韻を残すこともある。日本語は、語順／文節の出現位置の転換が自由だと言われる。確かに、「名詞＋助詞」あるいは副詞間の出現位置の転換は自由度が高いが [4]、他の文節の出現位置は必ずしも自由ではない。文末の文節については、特に、語順の転換の自由度が低いと思われる。従って、文脈自由文法の枠組でも対処できるであろう。

5.1 話しことばにおける文末の品詞

文末表現の多様性を把握するため、文末に使われている単語の品詞を対話データベースから検索した。対象は、キーボード会話 4674 文 (171 会話)、電話会話 6910 文 (151 会話) である。いずれも、国際会議に関する問い合わせをドメインとしている⁹。キーボード会話では、句点を一文の終わりとし、電話会話では、一人の話者の意図が、あるひとまとまりになったところを文と認定している。

⁹対話コーパスは随時追加されている。そのため、検索時期により、対象会話数が異なる。

表 2: 文末の品詞

品詞	キーボード会話		電話会話	
	出現頻度	%	出現頻度	%
助動詞	2679	58	2551	48
補助動詞	367	8	288	5
動詞	18	0	28	1
終助詞	1090	23	1237	23
接続助詞	232	5	712	13
係助詞	10	0	12	0
副助詞	7	0	11	0
格助詞	5	0	37	1
並立助詞	3	0	3	0
間投詞	182	4	403	8
副詞	37	1	39	1
接尾辞	10	0	12	0
数詞	6	0	14	0
固有名詞	2	0	12	0
普通名詞	2	0	14	0
形容詞	4	0	2	0
接続詞	3	0	2	0

表2から、書きことばでも、話しことばでも、助動詞と終助詞の使用率が高いことがわかる。一方、接続助詞については、キーボード会話では、出現率が5%に留まっているのに対して、電話会話では、13%と高くなっている。このことから、文末に接続助詞を使うのは、話しことばの特徴だと言えるであろう。

話しことばに関する研究、文献[13]では、接続助詞終止は省略による不完全なものとして、対象外にしているが、本稿では実データの検索結果を重視し、話しことばに多い接続助詞終止を認める。

5.2 構文規則の文節音声認識結果への適用

文節の音声認識傾向を調べるため、文節を基に文を組立てる規則を約50用意した。対話コーパス検索結果を考慮し、文末の品詞としては、助動詞、補助動詞、終助詞、接続助詞などを認めている。接続助詞としては、対話コーパス形態素作業マニュアル[14]を基に以下の単語を扱っている:

けれども、けれど、けども、けど、が、と、から、し、なら、ので、んで、の
に、て、ちゃ、たら、で、じゃ、だら、ば

これらの組立規則を文節音声認識結果¹⁰に適用した。音声認識対象は、国際会議に関する問い合わせ2会話37文である。37文中、接続助詞終止のものは3文であり、いずれにも「が」が使われていた。現在、文節音声認識には、2つの異なるHMM (Hidden Markov Model) 音韻モデル、Basic-モデル(以下B-モデル)とPrecise-モデル(以下P-モデル)¹¹が用意されているが、ここではB-モデルを使っている。実験の結果、文候補になった例を示す:

1 : もしもし

1a: moshimoshi

1b: oshie-ru-shi

1c: mo-chi-mashi-te

2 : そちらは会議事務局ですか

2a: sochira-wa kaingizimukyoku-desu-ka

2b: sochira-wa kaingizimukyoku-desu-kara

2c: kochira-wa kaingizimukyoku-desu-nga

3 : それでは登録用紙をお送り致します

3a: soredewa tourokuyoushi-o

ookuri-ita-shi-masu

3b: soredewa tourokuyoushi-mo

ookuri-ita-shi-masu-shi

3c: soredewa tourokuyoushi-o

ookuri-ita-shi-masu-to

1b, 1c, 2b, 2c, 3b, 3cでは、接続助詞が文末で使われている。その中で、1b, 1c, 3b, 3cは、容認性が低い文である。構文規則では、これらの文で使われている接続助詞が、文末で使われないように制約を加えるべきであろう。

¹⁰この実験の設定は、グローバルビーム幅100、一本の枝からのローカルビーム幅10である。

¹¹B-モデルでは、約35の音韻モデルを用意している。一方、P-モデルでは、語頭、語中などの区別をし、約70の音韻モデルを用意している。

表 3: 音声認識接続助詞終止

接続助詞	B- モデル		P- モデル	
	出現頻度	%	出現頻度	%
し	43	57	29	42
が	13	17	15	22
ちゃ	10	13	4	6
と	5	7	3	4
から	3	4	8	12
て	1	1	2	3
なら	0	0	5	7
たら	0	0	3	4

5.3 音声認識における文末の接続助詞

音声認識において誤認識の要因となりやすい接続助詞を別に扱うことを考える。文節音声認識の結果第5位までに入った文末文節候補185を対象に、文末の接続助詞を調べた。ここでは、HMMを使った認識傾向を調べるため、B-モデルとP-モデルを使った結果を比較する。B-モデルを使った場合75文節(41%)、P-モデルを使った場合69文節(37%)で、文末に接続助詞が使われていた。その内訳を表3に示す。

両モデルとも、「し、が、ちゃ」などの出現率が高い。これらの接続助詞は、音韻数の少ない単語なので誤認識の要因になり易く、出現率が高くなっていると思われる。特に「し」では、母音の無声化が作用していると思われる。一方、音韻数の多い「けれども」「けれど」などは、誤認識の対象にはなっていなかった。

5.4 対話コーパスにおける文末の接続助詞

実際の会話で、どのような接続助詞が文末で使われているか調べる。対話コーパスで使われている文末の接続助詞の内訳を表4に示す。

表 4: 文末の接続助詞

接続助詞	キーボード会話		電話会話	
	出現頻度	%	出現頻度	%
が	197	85	274	38
ので	11	5	96	13
て	8	3	23	3
で	0	0	1	0
から	6	3	14	2
けれども	5	2	212	30
けれど	1	0	18	3
けど	1	0	12	2
けども	0	0	37	5
し	2	1	10	1
もので	1	0	0	0
たら	0	0	5	1
と	0	0	2	0
ば	0	0	2	0
んで	0	0	5	1
ながら	0	0	1	0

キーボード会話では、「が」の使用が85%と顕著である。書きことばだけを対象とする場合には、文末で使われる接続助詞としては、「が」を認めるだけでよいであろう。しかし、話しことばでは、「が」と並んで「けれども」とそれに準じる単語の出現率も高くなっている。また、「ので」の使用率も、書きことばと比較してかなり高い。

前述の文献[13]では基本的に接続助詞終止を対象外にしているが、いくつかの文末の接続助詞は、例外的に認定することを明記している:

- それだけで理由を表すもの: ... だし, ... だから
- 曖昧、婉曲な終止を表すもの: ... だけど, ... したりして, ... しなきゃ

しかし、表4から明らかなように「し、から、けど、て」などの出現率はあまり高くない。従って、これらの接続助詞を文末に許さない構文規則でも被覆率はそれほど低くならない。特に「し」は表3に示すとおり音声認識において誤認識の要因となり易いので検討を要する。

6 構文規則の改良とその効果

3節の考察から、形式名詞、数詞、格助詞「へ」、接続助詞を含む文節が音声認識された文の容認性を下げる要因になりやすいことがわかった。そこで、4節と5節での検討を踏まえて文の構文規則を改良した。今回報告する文の構文規則では、この4点に以下のように対処している。

- 形式名詞は、必ず修飾を受けるものとして扱う。
- 数詞は、住所、電話番号、日付および値段としてのみ使用できる。数詞は音韻数が少なく特に誤認識されやすいので、強い制約を与える必要がある。我々は、国際会議に関する問合せをドメインとして扱っているので、そのドメインでの運用を考慮し、トラップをする状況などは扱わない。
- 格助詞「へ」を含む文節は、方向性を持たない述語とは共起を許さない。
- 接続助詞は、文末でも使われるものと、文中でのみ使われるものとを区別した。

この構文規則では、以下の6種類を文として認めている。

1. 「はい」「いいえ」などの感動詞を扱うもの。
2. 「それでは、登録用紙をお送り下さい」のように、接続詞が文頭にあり、述部で終るもの。
3. 接続詞が文頭にあり、述部で終るもののうち、述語が方向性を持たないもの。
4. 接続詞がなく、述部で終るもの。
5. 接続詞がなく、述部で終るもののうち、述語が方向性を持たないもの。
6. 例外を含むもの。

音声認識のための構文規則には、対話コーパス検索結果が直接反映されることが望ましい。例外規則では、音声入力 of 文節と、対話コーパスで定義している文節に食い違いがあるものを扱っている。

文節内規則と、文の構文規則の内訳を表5に示す。

表 5: 構文規則の規模とパープレキシティー

	文節内規則	文の規則
規則数	1,973	471
終端数	744	133
パープレキシティー	3.57 / 音韻	99.7 / 文節

改良前の規則の許容率が約 80% だったのと比較すると、この構文規則は同じ実験対象、353 文節からなる 137 文に対して、約 30% の許容率に下がった。次に、改良を加えた規則の、制約としての効果と、音声認識への貢献を調べる。許容率の変化を調べることにより、制約としての効果を見る。本稿では、制約としてもっとも効果があらわれていると思われる接続助詞について調べる。点線は、すべての接続助詞が文末に使われることを許した時の許容率であり、実線は、文中でだけ使われるものと、文末でも使われるものを区別した時の許容率である。一文中の文節数と許容率の相関関係を図 1 に示す。

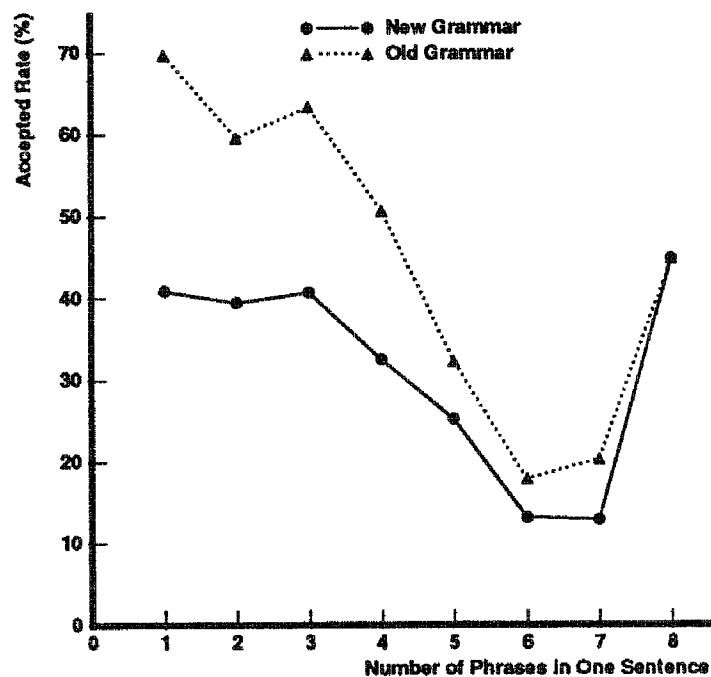


図 1: 一文中の文節数と許容率

図 1 から、構文規則の条件の強さにかかわらず、文節数が多いほど制約効果があることがわかる。特に、6 文節からなる文に対しては、両規則ともほぼ同じ制約効果を出している。一方、文節数が少ないものに関しては、制約の強い規則の方が、その効果が顕著にあらわれている。8 文節からなる文は、1 例だけなので、この値は信頼性に欠ける。

音声認識の誤認識傾向を考慮して改良した構文規則が、どの程度認識率の向上に役立っているか調べた。制約を使わず、文節候補に対してすべての組合せを、その尤度の順に並べた場合と、構文規則を制約として使った場合との累積文認識率の比較を表 6 に示す。

表 6: 構文規則の有効性

ランク	累積文認識率 (%)	
	制約なし	制約あり
1	37.2	70.1
2	60.6	76.6
3	68.6	83.9
4	70.1	83.9
5	73.7	83.9

表 6 から、構文規則の制約としての有効性がわかる。特に、第 1 位ではその効果が顕著である。

7 おわりに

対話コーパスから検証された、実際の言語使用を反映し、かつ音声認識傾向を考慮して、文節に基づく構文規則を構築した。また、その制約としての有効性を確認した。

現在、構文規則では、接続詞の文中での使用は考慮していない。話し言葉では、「それでは、登録用紙をお送り下さい。」の文節の順番を変えて、「登録用紙を、それでは、お送り下さい。」と言うこともある。また、名詞文節は、時を表し副詞的に働くものなど一部のものを除いて助詞を必ず伴うことを前提としている。しかし、「登録用紙お送り願えますか？」のように助詞を伴わない名詞文節もある。さらに構文規則を改良するために、接続詞の割り込みや助詞を伴わない名詞文節などについて、その実際の使用を対話コーパスを基に調べる予定である。

国際会議に関する問い合わせをドメインとする対話コーパス検索結果は、これまでの話しことば研究とのくいちがいを示唆している。話しことばの特徴を明示するにはさらに他のドメインについても調査する必要がある。

謝辞

本研究の機会を与えて下さると共に適切な助言を頂いた ATR 自動翻訳電話研究所樽松明社長、同データ処理研究室森元逞室長に感謝します。

参考文献

- [1] 竹沢, 森元, 樽松 1991: “日英音声言語翻訳実験システム *SL-TRANS 2* における音声対話処理”, 1991 年電子情報通信学会秋季全国大会.
- [2] 北, 坂野, 保坂, 川端 1989: “*SL-TRANS* における文節音声認識”, 第 39 回情報処理学会全国大会.
- [3] 尾関和彦 1988: “多文節間の係り受け整合度に基づき最適文節列を選択する多段決定アルゴリズム”, 信学論, **J71-D**, 4, pp. 669-677.
- [4] 久野すすむ 1973: 日本文法研究, 大修館書店.
- [5] 柿ヶ原, 森元 1989: “*SL-TRANS* における文節候補の削減 - 係り受け関係を用いた文節候補選択 -”, 第 39 回情報処理学会全国大会.
- [6] 江原, 小倉, 森元 1990: “電話対話データベースの構築”, 第 40 回情報処理学会全国大会.
- [7] 橋本, 小倉, 江原, 森元 1990: “対話データベースを用いた各種言語現象の検索”, 第 40 回情報処理学会全国大会.
- [8] 保坂, 小暮, 小倉 1990: “音声認識のための連鎖制約としての文法”, 第 40 回情報処理学会全国大会.
- [9] 竹沢, 保坂, 北, 森元, 江原 1990: “構文規則を用いた文音声認識”, 信学技報, SP90-73/NLC90-45.

- [10] 保坂, 竹沢, 江原 1991: “誤認識傾向を考慮した音声認識のための構文規則”, 第42回情報処理学会全国大会.
- [11] 小川, 林 1982, 1988: 日本語教育事典, 大修館書店.
- [12] 保坂, 竹沢, 江原 1990: “話し言葉における接続助詞終止とその音声認識への応用”, 第41回情報処理学会全国大会.
- [13] 国立国語研究所 1960: 話しことばの文型 (1), 秀英出版.
- [14] 篠崎, 水野, 小倉, 吉本 1989: 形態素情報利用解説書, *ATR* テクニカルレポート, TR-I-0077.

1 はじめに

連続音声の中から、最適な単語列だけではなく、適切な意味解釈を取り出すための汎用的な音声言語システム (Spoken Language System) の研究を進めている。音声言語システムの研究では、(1) 音声処理と言語処理のインターフェースをどのように考えていくか、(2) spontaneous な音声現象をどのように扱うか、(3) プロソディ等の音声固有の特徴を文の解析や談話処理のレベルにどのように反映させていくか、(4) 言語モデル、談話モデルとして、どのような枠組を用意するか、そしてそれをどのように音声レベルのサーチに反映させていくか、等の興味ある問題が当面の研究課題として考えられている。

本稿は、主に、(1) の問題に対して検討を行なったもので、連続音声の中から文脈自由文法、およびより一般的な句構造文法などの制約の下で、適切な単語列とその単語列に対する意味解釈を取り出すための基本的なアルゴリズムを示す。

ここで提案する手法は、従来の階層型のシステムで用いられていたワードラティスやフレーズラティスなどを中間表現として用いることなく、音声のフレームレベルの特徴系列から直接に文の候補やその意味解釈を取り出すことができる。そのため、階層間での情報の受け渡しにおけるオーバーヘッドを回避でき、談話レベルから得られる様々な制約を直接、音声のフレームレベルでの探索に反映させること可能である。同様の観点から、Ney [1] は C.K.Y. 法に基づく連続音声認識アルゴリズムを示している。この Ney のアルゴリズムは、(1) 基本的に音声の分析フレーム数 (フレームのシフト幅を 5msec 程度とし、5 秒の連続音声进行分析したとすると、フレーム数は約 1000 フレームとなる) の 3 乗に比例したオーダの計算量を必要とする、(2) 制御構造が基本的にボトムアップであり、すべての語彙や統語規則に対して網羅的な探索を行なう、などの問題点が指摘されている。

ここで提案するアルゴリズムは、One-Pass サーチ [2]、Viterbi サーチなどの音声のフレームレベルでの探索アルゴリズムを直接、文脈自由文法などのより一般的な制約の下で制御する枠組であり、音声のフレーム数に対してリニアオーダの処理となる。また、フレームに同期した探索を行なうなどの優れた性質を持っている。

以下、本稿では、(1) 現在、NTT 基礎研究所で構築を進めている音声言語システムの概要、(2) One-Pass サーチアルゴリズムを文脈自由文法で制御するための基本的な拡張、(3) One-Pass サーチアルゴリズムを動的に制御するための、チャート解析法に基づいたミニマルなネットワークの生成法、等について述べ、最後に、音声言語パーズングにおける探索と解釈の二つの過程、すなわち連続音声の中から最適な単語列を探索する過程と、単語列に対して統語的な構造や意味的な解釈を取り出す過程、相互の分離可能性について考察する。

2 音声言語システムの概要

ここ数年、米国の DARPA の音声認識に関する研究プロジェクトのターゲットが、不特定話者を対象とした大語彙連続音声認識の研究から、「音声言語システム (Spoken Language

System)」と呼ばれる、音声と自然言語処理とを“密に”統合したシステムの研究に移行しており、その動向が注目される [3]。これらのシステムでは、従来の Dictation Machine などを意図した大語彙連続音声認識のように、連続音声から最適な単語列を取り出すだけではなく、その単語列に適切な意味解釈を与えたり、談話レベルの制約を音声の探索に直接反映させるための基本的な枠組を必要とする。特に、spontaneous な音声を扱うためには、より能動的な枠組を必要とし、発話の生成過程に関する深い理解が不可欠である。例えば、spontaneous な音声を発話をする際のプランニングのモードは、written language のそれとは異なり、この違いが ill-formed な発話文を解析する際のポイントになると考えられる。

ここでは、素性構造の単一化に基づくパーシングアーキテクチャと One-Pass サーチアルゴリズムとを統合することにより、音声の探索と言語処理の探索アルゴリズムが“密な”結合関係にある統合アーキテクチャを実現している。素性構造の単一化の機構は、形態レベル、統語レベル、談話レベルやプロソディ等の様々な制約を宣言的、かつ統一的に扱うことができる。また、断片的な情報の扱い、制約を bi-directional に伝搬させるための機構、意味解釈を取り出すための機構など、幾つかの優れた性質を兼ね備えている。これは、次発話の予測に関する断片的な情報や動的に変化しうる様々な制約を扱うための基本的な機能を提供する。

また、様々な制約を音声のフレームレベルでのサーチアルゴリズムに反映させるために、ここでは拡張されたアクティブチャートパーザーが中心的な役割を果たしている。これは、より一般的な制約から、発話の解析と同時に One-Pass サーチのプロセスを制御するためのネットワークをインクリメンタルに供給していく汎用的な枠組になっている。システムのトータルなパフォーマンス (認識精度および効率性) を向上させるためには、様々な制約によって探索空間を最小化し、それを反映したミニマルなネットワークを生成することが重要なカギとなる。

3 単語列の探索アルゴリズム

音声処理と言語処理とのインターフェースの考え方として、単語や句などのラティスなどを中間的な表現として用いるラティスパーシングのアプローチがある [4-5]。また、日本語を対象とした連続音声認識システムでは、文節内と文節間で文法の表現方法や処理アルゴリズムを積極的に分離しようとするものが多い [6]。これらの方法の問題点は、(1) 単語列としての最適性が十分に保証されない、(2) 言語的な制約が単語列の認識、サーチのレベルに反映されにくい、(3) 同様な処理を階層間で重複して行なう、などの幾つかのオーバーヘッドの存在である。

ここでは、単語列の最適性が保証されている連続単語認識アルゴリズム (例えば、One-Pass サーチ法 [2]) を文脈自由な句構造文法、および単一化文法などのより一般的な制約に基づいて制御する方法を示す。この方法では、One-Pass サーチ法を制御するネットワークについて、音響的な整合度の高いアクティブな部分 (ここでは、アクティブゾーンと呼ぶ) だけに焦点をあてたサーチを行なう。これは、ネットワークの各エッジそのものに対する枝刈りとは本質的に異なり、ネットワーク中のエッジは決して削除されない。ネットワーク上の焦点をあてる部分、すなわちアクティブゾーンをフレーム同期型のビームサーチの手法 [7] を用いて、データ駆動的に遷移させていく。次に示すアルゴリズムは、One-Pass サーチアルゴリズムを、チャート解析法によって生成されたネットワークにより制御するための

手続きを示したものである。

Initialization of One-Pass algorithm:

```

Chart ← Make_chart_system(Start_symbol);
for each outgoing lexical edge  $LE_{k,1}$  at initial vertex do
   $w_{k,1} \leftarrow \text{Lexical\_item}(LE_{k,1});$ 
  for each frames  $j$  of  $w_{k,1}$  do
     $D(j, k, 1) \leftarrow \text{Initial\_cumulative\_score}(j, k, 1);$ 
     $B(j, k, 1) \leftarrow 0;$  /* Backpointers */

```

Frame-synchronous Process :

```

for input speech frame  $i = 1$  to  $N$  do
  for each vertex  $v \in \text{SubNetwork}$  do
    for each outgoing lexical edge  $LE_{k,v}$  do
       $w_{k,v} \leftarrow \text{Lexical\_item}(LE_{k,v});$ 
       $(AZ_s(k, v), AZ_e(k, v)) \leftarrow \text{Active\_zone}(LE_{k,v});$ 
      for  $j = AZ_s(k, v)$  to  $AZ_e(k, v)$  do
        if start_frame?( $j, w_{k,v}$ )
          then /* between-unit transition rule */
             $k' \leftarrow \{\text{incoming lexical edges at } v\};$ 
             $D_i(j, k, v) \leftarrow \max_{k'} \{D(L(k', v), k', v)\};$ 
             $B_i(j, k, v) \leftarrow i - 1;$ 
          else /* within-unit transition rule */
             $j^* \leftarrow \text{argmax} \{D(j - 1, k, v); D(j - 2, k, v)\};$ 
             $D_i(j, k, v) \leftarrow D(j^*, k, v) + d(i, j, k, v);$ 
             $B_i(j, k, v) \leftarrow B(j^*, k, v);$ 
        {end of loop}
      if active_edge?( $LE_{k,v}$ )
        and  $AZ_e(k, v) = L(k, v)$  then
          Edge ← Create_passive_edge( $LE_{k,v}, \text{Chart}$ );
          Add_item_to_agenda(Edge, Agenda);
      {end of loop}
       $k \leftarrow \{\text{incoming lexical edges at } v\};$ 
       $U(i, v) \leftarrow \text{argmax}_k \{D_i(L(k, v), k, v)\};$  /*unit*/
       $V(i, v) \leftarrow \text{start vertex of } U(i, v);$  /*vertex*/
       $F(i, v) \leftarrow B_i(L(U(i, v), v), U(i, v), v);$  /*frame*/
      SubNetwork ← Update_chart(Chart);
    {end of loop}
  for all  $j, k, v$  do
     $D(j, k, v) \leftarrow D_i(j, k, v); B(j, k, v) \leftarrow B_i(j, k, v);$ 
  {end of loop}

```

Traceback Process:

```

Solutions ← Trace_back(Chart, U, V, F);

```

One-Pass サーチプロセスでは、入力音声のフレームに同期して、チャートの中の特に語彙エッジから構成される遷移ネットワークにガイドされながら、探索を進める。ここで、 $D_i(j, k, v)$ および $B_i(j, k, v)$ は、ネットワークの頂点 v から分岐する k 番めの単語（あるいは音節、音素等の適当な単位）の j フレームに対応する格子点における、累積尤度とバック

ポインタである。また、 $D(j, k, v)$ および $B(j, k, v)$ は直前のフレームでの値を保持するためのテンポラリなバッファである。各入力音声のフレーム i において、ネットワーク上のアクティブゾーン ($AZ_s(k, v) \sim AZ_e(k, v)$) での累積尤度を計算し、関数 `Active_zone` は、その値から次フレームでのアクティブゾーンを推定する。もし、活性な語彙エッジの終端にすでにアクティブゾーンが到達していれば、関数 `Create_passive_edge` は、チャート解析のプロセスに対して、新たな非活性の語彙エッジとして提案する。この提案は、非同期で行なわれる。一方、チャート解析のプロセス (`Update_chart`) では、あらたに提案されたエッジに基づいて、完了操作、予測操作を行ない、チャートの内容を更新する。次のフレームでの処理は、この新たに更新されたチャートを参照して行なう。

$U(i, v)$ 、 $V(i, v)$ 、および $F(i, v)$ は、サーチが終了した時点で、最適な単語列をトレースバックして探し出すために用いられるポインタであり、詳細は省略する。単位内の遷移ルール (within-unit transition rule) は、通常的时间軸の非線形整合を行なうための傾斜制限であり、One-Pass サーチアルゴリズムでは単位間の遷移ルール (between-unit transition rule) が中心的な役割を果たす。このルールは、ネットワーク中に合流点がないと全く機能しないことに注意する必要がある。

個々の単語の標準パターン (あるいは、単語モデル) は、音韻環境に依存したセグメントを連結することにより構成するため、語彙記述の変更によって基本的には任意の語彙を扱うことができる。ここでは、DP に基づいた定式化を行ないアルゴリズムの概要を示したが、HMM では、Viterbi サーチを用いることでアルゴリズム、枠組に大きな変更はない。このアルゴリズムでは、単位間の遷移ルールを示すネットワークをインクリメンタルに展開していくための枠組が必要になる。つぎに、最小のネットワークを用意するためのチャート解析法を示す。

4 音声言語のパーズング

入力音声の発話内容に適応させてミニマルなネットワークを生成するためには、サーチアルゴリズム側からのデータ駆動的な情報に加え、チャートの制御方略が重要な要素となる。チャート解析法 [8] は、柔軟で効率のよい汎用的なパーズングの枠組として自然言語処理の分野で広く利用されている。音声言語に対するパーズングでも、不適格 (ill-formed) な入力や曖昧性の高い入力から、発話の内容を解析するためにチャートのデータ構造や制御方略が幾つか検討されてきた [5][9-10]。

Thompson [9] は、ラティスで表現された単語系列から、動的計画法の方略を用いて、スコアの高い (いわゆる、N-best な) 単語系列を捜し出す汎用性の高いチャート解析法を提案している。本稿で述べるチャート解析法は、この Thompson の枠組を、音声のフレームレベルに拡張したものと考えることができる。すなわち、ラティスなどの中間的な表現を介さず、直接フレームレベルから最適な単語系列とその統語構造を取り出す。また、談話レベルから得られる様々な制約をフレームレベルのサーチ (One-Pass サーチ) に反映させることができる枠組になっている。

4.1 ミニマルなネットワークを与えるアクティブチャート解析法

チャート解析法は、解析の部分結果を表 (WFST) を用いて管理し、重複のない効率的な解析をする。(a) 解析開始の段階で、あらかじめ単語の候補がデータ駆動的に用意されてい

ない、(b) 複数の可能性 (alternatives) を並行して調べ、相対的な比較操作により、最適パスを探索していく、という音声言語のパーシングの性質を考慮すると、基本的に下降型で横型の解析方略となる。アルゴリズムの概要を以下に示す。チャートの初期化では、アジェンダ (Agenda) の中身をクリアし、開始記号が S で始まるルールに相当するエッジをアジェンダに加える。

Initialization of Chart:

```
Clear_agenda(Agenda);
for each rule  $S \Rightarrow \alpha_1 \alpha_2 \dots \alpha_n$  do
    Edge  $\leftarrow [S \Rightarrow \cdot \alpha_1 \alpha_2 \dots \alpha_n, 0, 0, \{\}];$ 
    Add_item_to_agenda(Edge, Agenda);
{end of loop}
```

ここで、各エッジを次のような4つ組によって表現する。

$edge = [Dotted-rule, i, j, \{n, m, l, \dots\}]$

ここで、 i, j はそれぞれ start vertex、end vertex である。また、 $\{n, m, l, \dots\}$ はオルタナティブである娘のエッジに対してリンクを張るためのポインタのリストを表す。

Main Procedure for Updating Chart:

```
while not Agenda = null do
    Edge  $\leftarrow$  Get_item_from_agenda(Agenda);
    case EdgeSourceType of
        Prediction : /* for predicted edge */
            if not Subsuming?(Edge, Chart) then
                Add_item_to_chart(Edge, Chart);
                Prediction_and_completion(Edge);

        Completion : /* for completed edge */
            EqualEdge
             $\leftarrow$  Search_EqualParentEdge (Edge, Chart);
            if EqualEdge = nil then
                Add_item_to_chart(Edge, Chart);
                Prediction_and_completion(Edge);
            else
                Append_pointer_list(Edge, EqualEdge);

        Insertion : /* for inserted lexical edge */
            Add_item_to_chart(Edge, Chart);
            Prediction_and_completion(Edge);
    {end of case}
{end of while}
```

チャート解析は、エッジをアジェンダから選択し、そのエッジを表 (Chart) に加え、予測操作 (prediction step) と完了操作 (completion step) と呼ばれる二つの手続きを進め、その結果得られたエッジを再度アジェンダに加えることを繰り返して行なう。アジェンダから取り出されたエッジを表に加える際に、重複したエッジを加えることを避けるためのチェック (Subsuming?) や、オルタナティブの検出 (Search_EqualParentEdge) が行なわれる。すなわち、完了操作の際に、すでに同様な機能を持った親のエッジが存在すれば、同様な親エッジを新たに加えることは避け、すでに存在する親エッジに、ポインタによってオルタナティブ

な娘エッジとして帰属させる。この、ポインタリストのアペンドの操作は、Append_pointer.liにより行なわれる。

One-Pass サーチアルゴリズムでは、動的計画法の性格から探索の途中段階で大局的な意味で最適な単語やその境界を決定することができない。従って、ここで設定される頂点は時間軸から独立したものとなる。そのため、Thompson のチャートのように探索の過程で生じるオルタナティブを同一の頂点に合流させることはできない。一方、回帰ルールによって生じる時間軸方向でのオルタナティブも通常のオルタナティブと同様に扱え、ミニマルなチャートを構成することができる。

Procedure Prediction_and_completion(*Edge*):

```

case Edge of
  [ $A \Rightarrow \alpha \cdot B\beta, i, j, \{o, p, \dots\}$ ] : /* for active edge */
    /* prediction step */
    for each rule  $B \Rightarrow \gamma_1 \dots \gamma_n$  do
       $Edge \leftarrow [B \Rightarrow \gamma_1 \dots \gamma_n, j, j, \{\}]$ ;
      Add_item_to_agenda(Edge, Agenda);
    /* completion step */
    for each [ $B \Rightarrow \gamma, j, k, \{m, n, \dots\}$ ] on Chart do
       $l \leftarrow \text{Create\_new\_vertex}(\text{Chart})$ ;
       $Edge \leftarrow [A \Rightarrow \alpha B \cdot \beta, i, l, \{k\}]$ ;
      Add_item_to_agenda(Edge, Agenda);

  [ $B \Rightarrow \alpha, i, j, \{o, p, \dots\}$ ] : /* for passive edge */
    /* completion step */
    for each [ $A \Rightarrow \beta \cdot B\gamma, k, i, \{m, n, \dots\}$ ] on Chart do
       $l \leftarrow \text{Create\_new\_vertex}(\text{Chart})$ ;
       $Edge \leftarrow [A \Rightarrow \beta B \cdot \gamma, k, l, \{j\}]$ ;
      Add_item_to_agenda(Edge, Agenda);
{end of case}

```

また、同一性のチェックは以下の手続きによる。

Procedure Search_EqualParentEdge (*Edge*, *Chart*):

```

case Edge of
  [ $A \Rightarrow \alpha \cdot BC, i, j, \{o, p, \dots\}$ ] : /* for active edge */
    if ExistOnChart? ( $[A \Rightarrow \beta \cdot BC, i, k, \{q, r, \dots\}]$ ) then
      return ( $[A \Rightarrow \beta \cdot BC, i, k, \{q, r, \dots\}]$ );
    else return(nil);

  [ $B \Rightarrow \alpha, i, j, \{o, p, \dots\}$ ] : /* for passive edge */
    if ExistOnChart? ( $[B \Rightarrow \beta, i, k, \{q, r, \dots\}]$ ) then
      return ( $[B \Rightarrow \beta, i, k, \{q, r, \dots\}]$ );
    else return(nil);
{end of case}

```

4.2 動作例

ある名詞句 (NP) が頂点 {4} から予測されているものとし、そこで展開されるチャートの解析過程をトレースすると次のようになる。

- (1): [$NP \rightarrow .AN, 4, 4, \{\}$]
- (2): [$A \rightarrow .\text{"aoi"}, 4, 4, \{\}$] ... predicted from (1).
- (3): [$A \rightarrow .\text{"akai"}, 4, 4, \{\}$] ... predicted from (1).
- (4): [$NP \rightarrow .N, 4, 4, \{\}$]
- (5): [$N \rightarrow .\text{"hana"}, 4, 4, \{\}$] ... predicted from (4).
- (6): [$N \rightarrow .\text{"hako"}, 4, 4, \{\}$] ... predicted from (4).

(1)～(6)までのステップは、通常のチャート解析法における予測操作と同様であり、この時点で4つの語彙エッジが予測される。

One-Pass サーチのプロセス(OP)では、サーチの進行に伴いアクティブゾーンを更新する。もし、このアクティブゾーンが語彙エッジの終端に到達した場合、この語彙エッジに基づいてチャートに対して非活性の語彙エッジを提案する。

- (7): [$A \rightarrow \text{"aoi"}, ., 4, 5, \{\}$] ... proposed from OP.
- (8): [$NP \rightarrow A.N, 4, 6, \{5\}$]
... combining (7) with (1).
- (9): [$N \rightarrow .\text{"hana"}, 6, 6, \{\}$] ... predicted from (8).
- (10): [$N \rightarrow .\text{"hako"}, 6, 6, \{\}$] ... predicted from (8).
- (11): [$A \rightarrow \text{"akai"}, ., 4, 7, \{\}$] ... proposed from OP.
- (12): Modifying (8)
to [$NP \rightarrow A.N, 4, 6, \{5, 7\}$] from (11).
- (13): [$N \rightarrow \text{"hana"}, ., 6, 8, \{\}$] ... proposed from OP.
- (14): [$NP \rightarrow AN., 4, 9, \{8\}$]
... combining (13) with (8).
- (15): [$N \rightarrow \text{"hako"}, ., 4, 10, \{\}$] ... proposed from OP.
- (16): Modifying (14)
to [$NP \rightarrow AN., 4, 9, \{8, 10\}$] from (15).

(7), (11), (13), (15)などが、One-Pass サーチのプロセスから提案されたエッジであり、このエッジに基づいて完了操作が行なわれる。完了操作において、新たに加えられる親エッジが、その要因(source)となった娘エッジに対するリンク情報を保持する。(12), (16)のステップでは、オルタナティブな娘エッジを共有化している。すなわち、すでに同様な機能を持った親のエッジが存在すれば、同様な親エッジを新たに加えることは避け、すでに存在する親エッジに、オルタナティブな娘エッジとして新たに加えておく。

これらのリンク情報に基づいて、語彙エッジを結びつけていくと、対応するネットワークが構成される。

5 インプリメンテーション

上述した枠組に基づく音声言語システムを構築した。このシステムは文節境界等をとくに意識せず連続に発声された音声を対象とする。素性構造の単一化の枠組で記述された制約の下で、最適な単語系列とその統語的、意味的な構造を抽出する。汎用的な枠組であり、特定のタスクに適用する際には、文法記述や語彙記述の枠組を変更するだけでよい。

システムは、最適な単語列を探索する One-Pass サーチプロセスと、この単語列のサー

チに対する統語的な制約を与えるための遷移ネットワークをインクリメンタルに生成する、チャート解析のプロセスとが非同期的に接続された構成となる。プロセス間での通信量は比較的少ない。インプリメントを行なう際には、これらのプロセスを幾つかの計算機に機能分散させることが考えられが、この時、プロセス間の通信によるオーバーヘッドは問題にならないほど小さい。処理時間等は、入力される連続音声や文法記述の内容によって大きく影響を受けるが、現在の実験の規模では、1文章の処理時間は数秒のオーダーである。

6 考察

6.1 インターフェースの考え方

ここでは、音声のフレームレベルでのサーチプロセスに対して、様々な言語的制約を反映した単語の遷移ネットワークをインクリメンタルに供給するという観点から、音声言語のパーズングの枠組をとらえた。また、サーチプロセスに対して「必要最小のネットワーク (minimal network)」を用意するという観点で、最適な単語系列のサーチとパーズング処理の効率の問題をとらえた。

このような“密な”統合関係にあるアーキテクチャでは、階層間でオーバーヘッドが回避でき、最適な解を得ることができる。反面、システム全体の処理が重くなりやすいという傾向がある。この問題を回避するために、大語彙を対象としたシステムでは、一般的な句構造文法などを有限状態ネットワークで近似する [11-12]、あるいは言語的な制約を二つに分離し、単語列の探索では比較的緩い制約を用いることが多い [13-14]。しかし、これらのアプローチでは、いわゆる言語的制約がフィルタリングとしてしか機能せず、談話レベルからの予測情報が有効に利用できない。ここでは、予測的な情報が反映できる枠組を残したまま、最適な単語列の探索とパーズングにおける解釈の過程を意識的に分離した枠組となっている。

6.2 探索と解釈フェーズの分離

連続音声をフレームレベルでパーズする際には、性質の異なる二種類の“曖昧さ”が存在する。すなわち、(1) 統語規則中に曖昧さが含まれる場合に、ある局所的な単語系列について、構造的に複数の解釈が可能な場合がある。この曖昧さは、他の談話的な情報や音声的な情報が付加されない限り、一意化されない。(2) 音声認識の不確実さを考慮すると、単語系列として幾つかの候補、あるいは可能性 (sub-sentence alternatives) を保持しながら探索していく必要がある。これらのオルタナティブは、One-Pass サーチにより最終的には最適な単語系列として一意化されるものであるが、探索の途中段階では最適な単語系列は知り得ない。

ここでは、インクリメンタルに生成されるミニマルなネットワークを介して、情報の流れを制御する。基本的には、チャート解析法により、各レベルで次のような方略が可能である。

1. 音声の探索におけるオルタナティブ (音声レベルでの曖昧性) の影響を可能な限り解釈レベルに反映させない。

オルタナティブのパス (単語の部分系列) に対して、独立に部分木を与えると、部分木の総数は、可能なパスの組合せ数 (指数オーダー) に膨れ上がる。

2. パージングの過程で生じる構造的な曖昧性の影響を単語列の探索レベルに反映させない。

ある局所的な単語系列について、構造的に複数の解釈が可能な場合がある。探索の際には、曖昧さを含んだ構造を可能な限りパックし、以降の解析に曖昧さをひきずらない。この探索が終了した時点で、曖昧な構造を再構築する枠組を用意することができる。

3. 素性構造のなかで、探索に関与せず、解釈結果だけを親に伝搬させる素性の具現化を遅らせる。

単語列の探索に部分的な意味解釈の結果を必要としないならば、膨大なオルタナティブに対する個々の意味解釈等は、この時点では必要でない。最適な単語系列が決定した後に、解析木の作成や、パッキングされた部分木の解凍、意味の解釈を行なってもよい。解釈の過程では、探索の過程で作られたチャートをそのまま利用することができる。

統合化された様々な制約に対して、確率的な重み付けを行なうことは難しいが、一度ネットワークにブレークダウンされるとネットワークの遷移確率として、幾

つかの情報を付与することが容易になる。また、我々が無意識に発話の解析を行なう際には、低いレベルでネットワークサーチのようなリニアオーダーの解析を行なっている可能性がある。高次レベルでの処理との切り分けの問題など、包摂アーキテクチャにも関連した興味ある問題がまだ残されている。

7 おわりに

音声のフレームレベルでのサーチアルゴリズムと統語処理におけるパージングアルゴリズムとを適切に融合する手法を示した。特に、チャート解析の枠組に基づいてミニマルなネットワークを動的に生成するための、具体的なチャートのデータ構造と制御方略を示した。チャート解析法は、LR 法と異なり動的な予測を行なう。そのため、発話の内容、対話の状況によって制約を動的に変更するような柔軟なパージングが行なえる。今後は、より自然な発話 (spontaneous speech) を扱うための制御方略を考えていきたい。

謝 辞

日頃、御討論いただく NTT 基礎研究所 島津 明 主幹員、竹内研究グループの諸氏、計量計画研究所 大塚 裕子 研究員に感謝いたします。

参 考 文 献

- [1] H.Ney : "Dynamic Programming Speech Recognition using a Context-free Grammar", Proc. ICASSP-87, pp.69-72 (1987).
- [2] J.S.Bridle, M.D.Brown and R.M.Chamberlain : "An Algorithm for Connected Word Recognition", Proc. ICASSP-82, pp.899-902 (1982).

- [3] 例えば、Proceedings of the DARPA Speech and Natural Language Workshop, Morgan Kaufmann (1989-02).
- [4] M.Tomita : “An efficient word lattice parsing algorithm for continuous speech recognition”, Proc. ICASSP-86, pp.1569-1572 (1986).
- [5] L.Chien, K.J.Chen and L.Lee : “An Augmented Chart Parsing Algorithm integrating Unification Grammar and Markov Language Model for Continuous Speech Recognition”, Proc. ICASSP-90, pp.585-588 (1990).
- [6] 岡田, 松尾, 伊藤, 小川, 牧野, 城戸 : “文節スポッティングに基づく日本語文音声の認識”, 電気学会論文誌 (C), Vol.108-C, No.10, pp.826-833 (1988-10).
- [7] H.Ney, D.Mergel, A.Noll and A.Paeseler : “Data-Driven Organization of a Dynamic Programming Beam Search for Continuous Speech Recognition”, Proc. ICASSP-87, pp.833-836 (1987).
- [8] M.Kay : “Algorithm Schemata and Data Structures in Syntactic Processing”, Technical Report CSL-80-12, Xerox PARC (1980).
- [9] H. Thompson : “Best-first enumeration of paths through a lattice – an active chart parsing solution”, Computer Speech and Language, 4, pp.263-274 (1990).
- [10] G.Satta and O.Stock : “Formal Properties and Implementation of Bidirectional Charts”, Proc. AAAI - 90, pp.1480-1485 (1990).
- [11] F. Pereira and R. Wright : “Finite-state Approximation of Phrase Structure Grammars”, Proc. of the 29th ACL, pp.246-255 (1991).
- [12] A. Black : “Finite-state Machines from Feature Grammars”, International Workshop on Parsing Technologies, pp.277-285 (1989).
- [13] R. Schwartz and Y-L. Chow : “The N-Best Algorithm: An Efficient and Exact Procedure for Finding the N Most Likely Sentence Hypotheses”, Proc. ICASSP-90, pp.81-84 (1990).
- [14] R. Pieraccini and C-H. Lee : “Factorization of Language constraints in Speech Recognition”, Proc. of the 29th ACL, pp.299-306 (1991).

A Study on the Roles of Prosody in the Cooperative Dialogue

Shin'ya Nakajima

(NTT Human Interface Laboratories)

James F. Allen

(University of Rochester)

1991 年 12 月 13 日

Abstract

This paper describes how well prosodic information correlate with the topic structure of discourse. To investigate this correlation systematically, first we introduce the notion of *utterance unit* which can be viewed as a basic unit in conversations. We then define four topic boundary classes; *Topic Shift*, *Topic Continuation*, *Elaboration*, and *Speech-Act Continuation*. The prosodic parameters—onset, first-peak, and final pitch frequencies—are measured at these topic boundaries to show how these prosodic parameters vary depending on the topic structure. Finally, we propose a schematic algorithm which identifies the topic boundaries via the prosodic parameters.

1 Introduction

The last decade has seen substantial progress in discourse processing and computational linguistic fields. Specifically, the plan recognition approaches based on Austin and Searl's speech-act theory [Austin 62, Searle 69] have been proposed (e.g. [AllPer 80]). However, although a number of analysts have pointed out that prosody plays several important roles in natural conversations (e.g. [BroYul 83, PieHir 90]), there have been very few studies that take account of the prosodic information. In general, the intentional meaning of the utterance in a conversation cannot be determined without referring prosodic information.

Prosodic information plays various pragmatic roles in a conversation; The most salient function of intonation is questioning. That is, by finishing a sentence with rising intonation, we can create a

yes-no question. Prosody can also specify the information structure—such as new/old information, and the topic structure. This paper focuses on the latter function of prosody, and we will show how prosodic information can be utilized as a cue for topic structure identification.

In the next section, we introduce our specific task domain—TRAINS world [AllSch 91]—and describe how we have collected natural conversations. We then define the topic structure markers which are based on the notion of *utterance unit*. Finally, we will show how well particular prosodic parameters correlate with the topic structure and propose a schematic algorithm which identifies the topic structure from the prosodic parameters.

2 Speech Data Collection

The map of the TRAINS world is shown in Fig.2-1. The cities in the TRAINS world are connected to each other by rail lines. Each city have either a manufacturing capability (OJ factory or beer factory), or storage capability. Transportation is supplied by engines, boxcars, and tankers which are initially placed at specific cities.

A user or Human (hereafter called *H*) should achieve a specific goal by making plans to manufacture and ship various goods to specified cities by the due date. Another person called System (*S*) has up-to-date knowledge on the state of the world and assists *H* in making plans to achieve the given goal.

While making plans, *S* and *H* are sitting in different rooms and communicate by using microphones and head phones. The speech of *H* and *S* is recorded on the right and left channel of digital audio tape. We collected a total dialogue duration

of about one and half hours from six goal-achieving sessions.

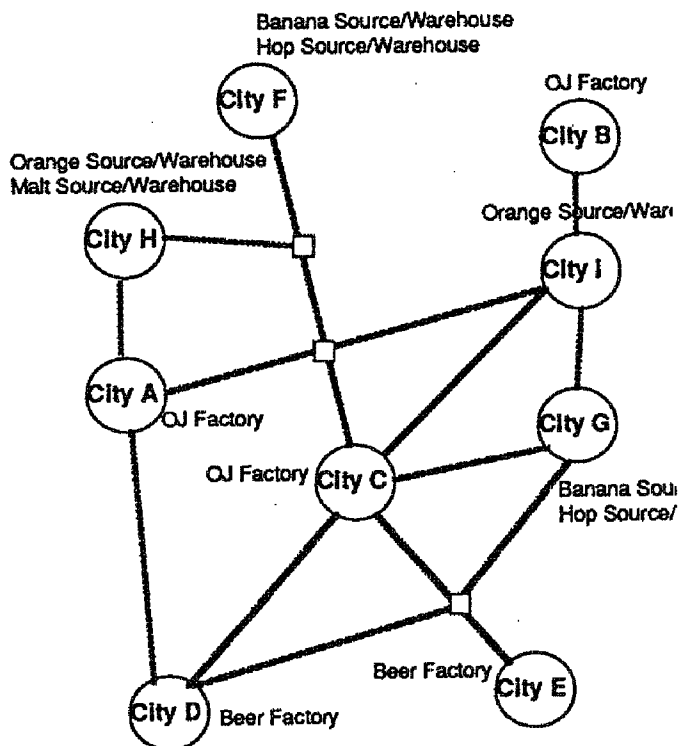


Fig. 2-1 The TRAINS domain for data collection

3 Discourse Structure Marking

3.1 Utterance Unit

Since grammatical units such as *sentences* are absent in the spontaneous conversations, we must first determine what is the basic unit of conversation to analyze the discourse structure systematically. We refer to this unit as the **utterance unit (UU)** which can be determined by following principles.

- **Grammatical Principle;** Place the UU boundary where a period could be put. In case of sentence conjunction, the UU boundary is set just before the conjunction.
- **Pragmatic Principle;** The UU should correspond to a basic speech-act. In other words, UU should represent the speaker's basic intention. Please note that this does not rule out the case where one speech act continues over several UUs. Actually, the utterance corresponding to a single speech act can be broken down to discrete UUs by the following two principles.
- **Conversational Principle;** A UU boundary should be placed whenever speaker changes.

This includes the case of short acknowledgement such as *hnn-hnn* or *yes*.

- **Prosodic Principle;** The UU boundary is placed whenever a medium length or longer pause occurs. The pause threshold is set to 750 msec which is a bit longer than the pauses called *search pauses* or *repair pauses*.

By applying these rules to the speech data, the utterances were split into numbered UUs. Ex.3-1 shows typical UU analysis. The utterance in Ex.3-1a is split into two UUs; the first UU, *okay*, is an acknowledgement, and the second UU is WH-question. Ex.3-1b shows the case in which S's acknowledgement *hnn-hnn* is inserted in the middle of H's statement.

H: okay, how long will it take for engine E3
to go to city I
↓

(H:uu1 okay)

(H:uu2 how long will it take for engine E3
to go to city I)

Ex.3-1a Utterance Unit analysis; including *okay*.

H: let's uhh move engine E3...

S: hnn-hnn

H: to city I
↓

(H:uu1 let's uhh move engine E3)

(S:uu2 hnn-hnn)

(H:uu3 to city I)

Ex.3-1b Utterance Unit analysis; acknowledgement.

The discourse structure and the prosody analysis discussed in the following sections are based on UU as defined. That is, the topic boundary variations are viewed as the relationships between the current UU and the previous UUs, and the prosodic parameters are measured for each UU.

3.2 Topic Boundary Types

To investigate the correlation between prosody and the discourse structure, we categorized the

topic boundary into four classes: **Topic Shift**, **Topic Continuation**, **Elaboration**, and **Speech Act Continuation**. These can be defined as follows. (Actual examples are shown in Ex.3-2.)

Ex. 3-2 Examples of each Discourse Segment Boundaries

A. New Topic

- 1 H: how many boxcars of oranges does it take to produce a tanker of oranges.. orange-juice
- 2 S: one boxcar uhh of oranges makes a boxcar.. a tanker of orange-juice
- 3 H: okay
- > 4 H: System, should I uhhh.. would you recommend that I uhh use my engine E3 to go to city I ?

B. Topic Development

- 1 H: is there orange-juice already made at city A ?
- 2 S: no, there's no orange-juice uhh made at all, right now
- 3 H: at all, at any of the cities ?
- 4 S: that's right
- > 5 H: how about uhh bananas, we have bananas at city F and G ?

C. Interruption

- 1 H: and I would like to brin...
- > 2 S: use E3 for that ?
- 3 H: yes

D. Topic Continuation

- 1 H: uhhh for beer I need uh hops and malt, is that correct ?
- 2 S: that's right
- 3 H: and I need a beer factory ?
- 4 S: yes, hnn-hnn

E. Elaboration

- 1 H: are there oranges available in ware houses in both cities H and I uhh let's see
- 2 S: there're oranges available in uhh yes, in H and in city I
- > 3 S: They have oranges in both places, enough for uhh uhm several boxcars of oranges

F. Clarification

- 1 H: let's do that
- > 2 H: let's move E2 to city E

G. Summary

- 1 S: actually, there's 20 tanker loads at D, I think
- 2 H: at D
- 3 S: and uhh something like thirty at E
- 4 H: E
- > 5 S: so plenty of beer

H. Speech Act Continuation

- 1 H: now let's uhh assume the oranges are already loaded into the boxcar B6
- 2 S: hnn-hnn
- > 3 H: and We'll take the engine that's at city H
- > 4 H: we'll move the boxcar with engine down to city A

Topic Shift (TS) This class can be viewed as three subclasses;

New Topic (NT) The current UU introduces a new topic. In our TRAINS domain, since S and H try to cooperate to achieve a particular goal, such utterances on new (sub)goal or new (sub)plan are taken as NT, rather than completely in-

dependent topics. In Ex.3-2a, after asking some questions, H introduces a new plan at utterance 4.

Topic Development (TD) The topic in the previous utterances is further developed at the current utterance and there might be some weak linkage between them. In Ex.3-2b, at utterance 5, H shifts his focus from the orange juice to the bananas, but there is a shared topic between them, namely, *search for resources involved in the goal*.

Interruption (Int) The previous or simultaneous utterance is interrupted abruptly by the current utterance. In Ex.3-2c, utterance 1 is interrupted by S's question.

Topic Continuation (TC) The linkage between the current topic and the previous one is comparatively strong. The current utterance may be talking about the same plan or the same entity as discussed in the previous utterance. In Ex.3-2d, at utterance 3, H continues to talk about *making beer*.

Elaboration Class (ELB) This class also can be viewed as three subclasses. The general interpretation of this class is that, the current utterance adds some relevant information to the previous utterance(s).

Elaboration (Elab) The current utterance adds some relevant information to the previous statement. In Ex.3-2e, S informs H of the quantity of the oranges which S believes relevant to H's last question.

Clarification (Clr) The current utterance clarifies some propositions involved in the previous utterances. In Ex.3-2f, H restates his proposal while clarifying what *do that* really means.

Summary (Summ) The current utterance summarizes the contents of the preceding utterances. as shown in Ex.3-2g.

Speech Act Continuation (AC) A single speech act continues over several UUs. Most of them

are sequential conjunctions as shown in Ex.3-2h.

In the following section, we describe how some prosodic parameters vary depending on the topic boundary classes and how the variation can be interpreted from the pragmatic viewpoint.

4 Prosody and Discourse Structure

4.1 Onset and First Peak Pitch Frequencies

A number of analysts have suggested that onset and first peak pitch are raised when the topic of the conversation is changed. (e.g. [BroCur 80]) However, to my best knowledge, clear and reliable confirmation has yet to be shown. In order to clarify how this prosodic tendency reflects on the topic boundary classes of our database where acknowledgements and interruptions are frequently made by the participants, we investigated the onset/first peak pitch frequency at each topic boundary class.

The measuring points of onset pitch (Po) and first peak pitch (Pp) are illustrated in Fig.4-1. For analysis consistency, we excluded the cases in which a single grammatical phrase (e.g. noun-phrase, prepositional-phrase, and so on) is split into several UUs via the prosodic principle. For instance, the cases like (H:uu1 from city...) [1 sec. pause] (H:uu2 G) were excluded. Since we are focusing here on the relationship between topic-shifting and onset/peak pitch, we also excluded simple answer utterances.

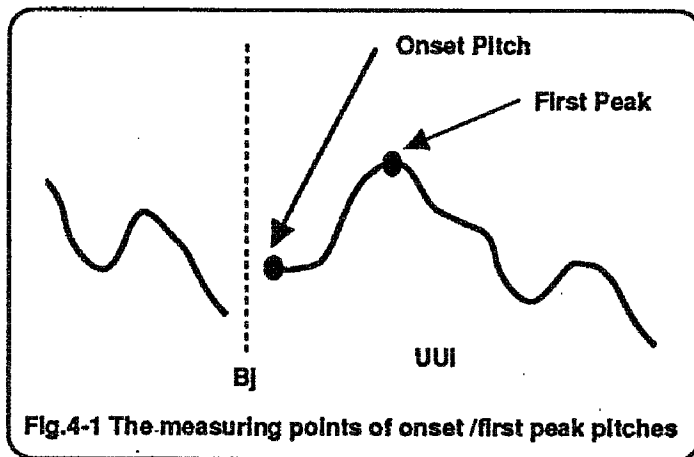


Fig.4-1 The measuring points of onset /first peak pitches

Onset/first peak pitch averages at each topic boundary class are shown in Fig.4-2. The results can be summarized as follows;

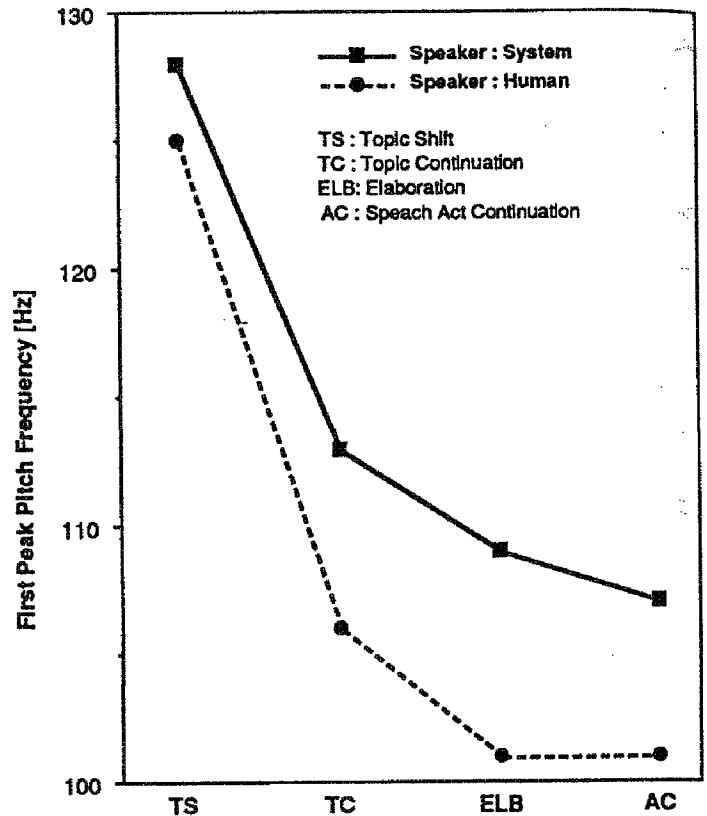


Fig. 4-2A Onset Pitch Frequency at each Topic Boundary

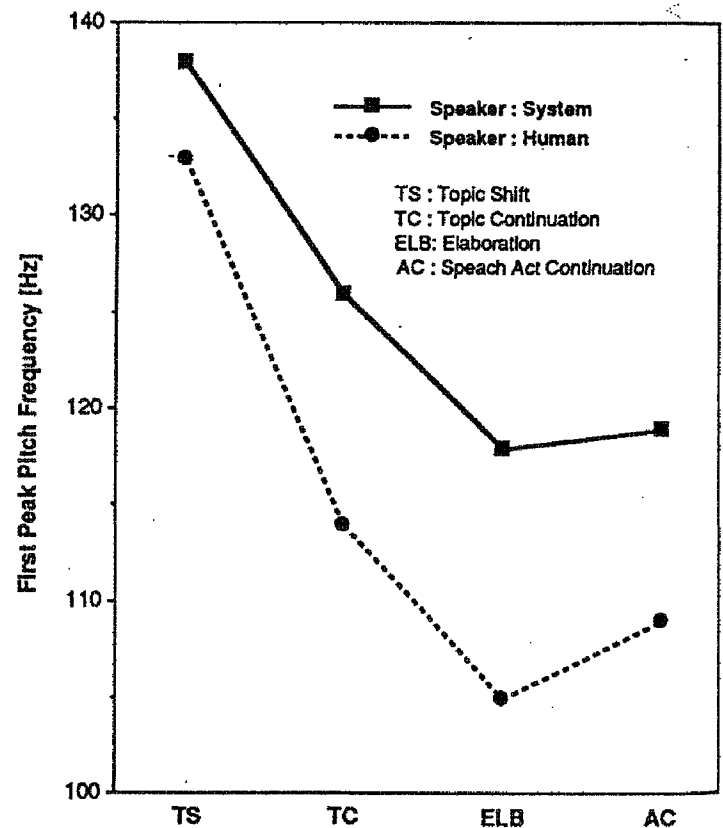


Fig.4-2B First Peak Pitch at each Topic Boundary

- For each speaker, both P_o and P_p decline in the order;

$$TS > TC > ELB \approx AC$$

For both speakers, the distinction between TS and other boundary classes is much more significant than the other differences.

- P_o/P_p at ELB boundary and those at AC boundary are almost identical for both speakers. This result suggests that as far as P_o and P_p are concerned, the prosodic connection between the previous and the current elaboration utterance is as strong as that of speech act continuation utterances.
- From the T-distribution tests, the statistical significance of P_o is higher than P_p 's for all cases. That is, onset pitch is a more reliable parameter than the first peak pitch, at least in terms of topic boundary class identification.

4.2 Final Pitch Frequency

As suggested in the literature, the final boundary tone reflects *finality* or *completeness* of the statement in declarative sentences. We investigated the correlation between final pitch frequency (P_f) and topic boundary class to show how this tendency is reflected in actual pitch contour.

The measuring point of P_f is illustrated in Fig.4-3. The final pitch of single answers, not followed by any subsequent utterances, are counted together with those of TS boundaries and referred as END class. This is because there is no significant distinction between the isolated answers and the topic shift boundaries.

The average of final pitch frequency at each topic boundary is shown in Fig.4-4.

As can be seen in the figures, for both speakers S and H, final pitch is much higher at AC boundaries than at other boundaries. Moreover, P_f s at boundaries other than AC are almost identical. Thus, final pitch frequency can be taken as a good cue for discriminating AC boundaries from other boundaries.

The previous results suggest that as far as onset and first peak pitch are concerned, the prosodic

connection at the elaboration boundary is as strong as that of speech-act continuation, whereas the final pitch result indicates considerable isolation between the previous and elaboration utterances. However, this phenomena can be explained by the semantic definition of elaboration class boundary and the pragmatic roles of prosody. At an elaboration boundary, the previous utterance UU_0 *per se* completes a particular statement, and the succeeding elaboration utterance UU_1 adds some relevant information to UU_0 . So, the completeness of UU_0 leads to the final pitch lowering and the following relevant utterance influences on the onset and first peak pitch values of UU_1 .

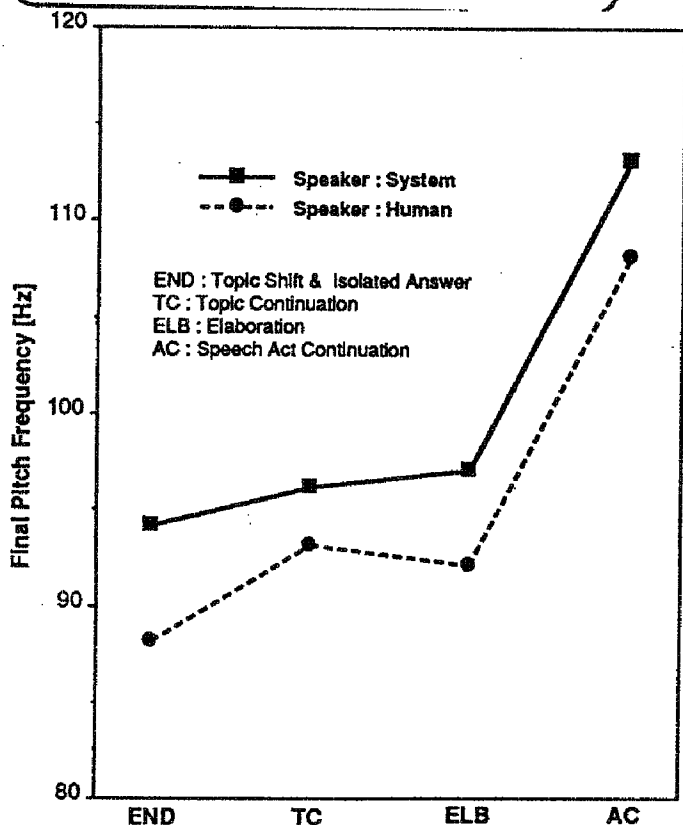
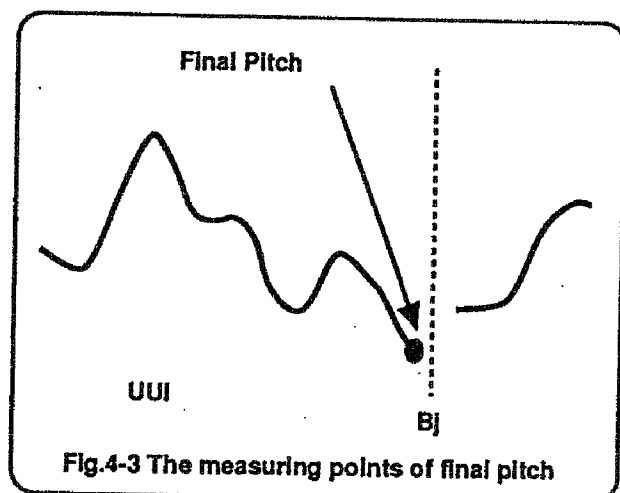


Fig. 4-4 Final pitch frequency at each topic boundary

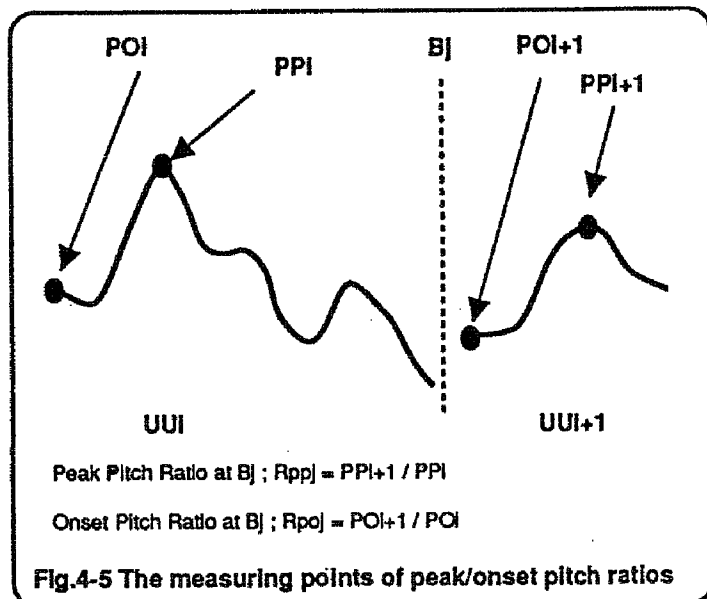
We'd like to note that when measuring the fi-

nal pitch frequencies, we do not discriminate rising tones from falling tones. Actually, however, while rising tones are the most typical pitch contours at AC boundary, we have found some so called *half completion* falling contours [Gussenhoven], where the pitch falls to mid-level. This fall can be also taken as indicating non-finality of the utterance.

4.3 Onset and First Peak Ratio

It is claimed that within a continuous speech, the peak pitch range of each intonational phrase declines towards the end of sentences [HakSat 80, LiePie 84, Ladd 84]. [HakSat 80] also suggested that as the grammatical connection between two neighboring phrases increases, the peak of the second phrase is suppressed more relative to the first phrase.

In this section, we extend the application of this tendency, from sentence speech to a sequence of linked utterance units, and show how this phenomenon is reflected in each topic boundary class.



To investigate the degree of declination, we use the ratio of the current UU's first peak pitch (or onset pitch) to that of the previous one. The measuring points are illustrated in Fig.4-5. As illustrated in the figure, both onset and first peak frequencies of the current UU₁ (Po₁, Pp₁) and the previous (same speaker's) UU₀ (Po₀, Pp₀) are measured. Then the declination ratios of onset pitch (Rpo) and first peak pitch (Rpp) at boundary B_j are computed as follows.

$$Rpp = \frac{Pp_1}{Pp_0}, \quad Rpo = \frac{Po_1}{Po_0}$$

We refer to the former as peak pitch ratio (Rpp), and the latter as onset pitch ratio (Rpo).

The averages of peak pitch ratio and onset pitch ratio are shown in Fig.4-6.

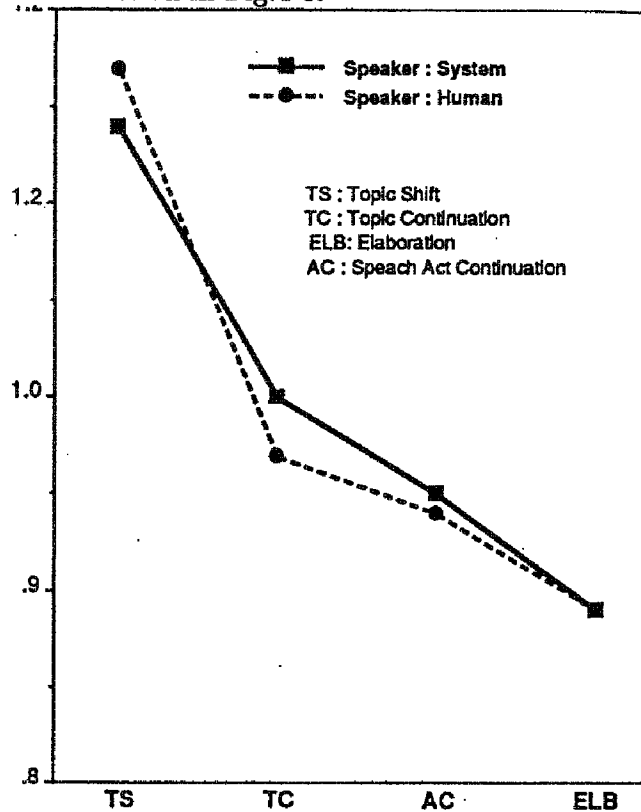


Fig. 4-6A First Peak Ratio at each Topic Boundary

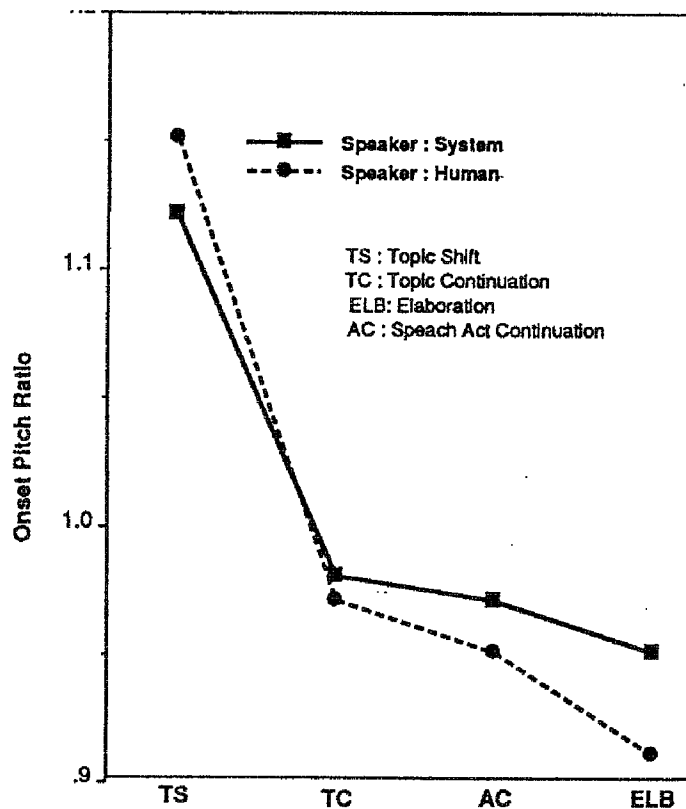


Fig. 4-6B Onset Pitch Ratio at each Topic Boundary

The results can be summarized as follows;

- For both speakers, the first peak ratio declines in the order;

$$TS > TC > AC > ELB$$

The onset pitch ratio also shows a similar tendency, but the distinction between the boundary classes other than TS is less significant than in the case of the first peak ratio.

- Both peak and onset pitch ratios are larger than 1.0 at TS boundaries. This result means that these parameters are raised at TS boundaries (about 1.15 times) relative to those of previous utterance. The peak pitch ratio at TC boundaries is around 1.0, so, this suggests that if there's no salient relationship and no abrupt topic shifting between two utterances, the speaker utters them with the same peak pitch range.
- For both speakers, both ratios (R_{pp} , R_{po}) at ELB boundaries are slightly lower than those at AC boundaries. This result can be interpreted as follows; the relationship between two utterances at an AC boundary is mostly coordinate, whereas elaboration utterances are sometimes subordinate to the previous ones. This subordination suppresses elaboration utterances more than coordination utterance.
- As can be inferred from Fig.4-6, the peak pitch ratio is a more reliable parameter than onset pitch ratio in terms of topic boundary identification. In other words, declination or suppression tendency is more salient on the *top line* than on the *base line*. Moreover, only the peak pitch ratio can discriminate ELB boundaries from TC boundaries reliably.

4.4 Topic Boundary Identification via Prosody

In this section, we discuss how our results can be utilized for topic boundary identification. From this point of view, the results shown above can be summarized as follows;

- Onset pitch is the best parameter to discriminate topic shift boundaries.

- Final Pitch is the best parameter to locate speech-act continuation boundaries.
- To discriminate elaboration boundaries from topic continuation boundaries, peak pitch ratio can be used.

These conclusions lead to the topic boundary discrimination tree described in Fig.4-7.

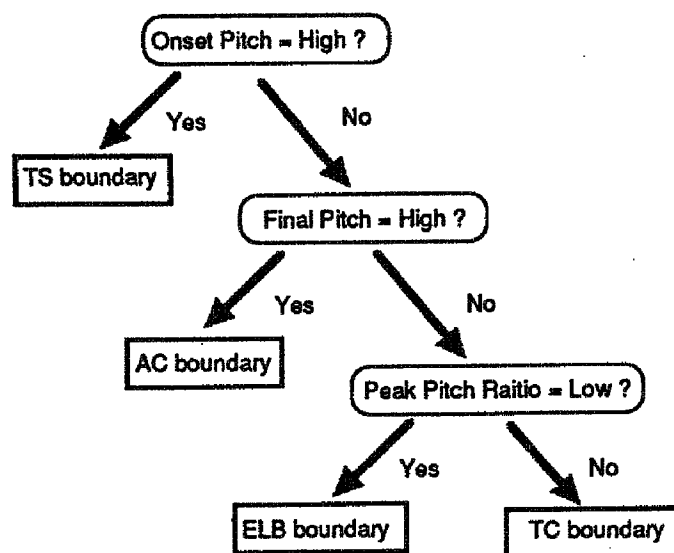


Fig.4-7 The topic boundary discrimination tree

5 Discussions

To develop a practical topic boundary discrimination algorithm, two problems must be overcome. First, as we have seen in the previous results, there is a considerable difference in pitch range depending on the speaker. Therefore, a sort of normalizing technique should be utilized to eliminate this effects. Another problem is that, since the prosodic phenomena described above reflect statistical effects, literal information should be also taken into account together with prosody. The following literal information will be useful in identifying the topic structure.

- Clue words; *okay, so, now, well*
If used with falling intonation, these clue words are often used as topic shift markers, and deaccented *so* is a good cue for indicating summarization.

- *Vocative; System*

In our speech database, vocative *System* is always used at topic shift boundaries

- *Form of question;*

Wh-questions are frequently used at topic shift boundaries, and declarative/tag-questions are normally used at topic continuation boundaries.

Thoroughly investigating such literal cues and showing how they can be used in combination with the prosodic cues are beyond this article, and left as a future task.

In this paper, we have been focusing on the correlation between prosodic information and the topic boundaries. However, there might be a more micro-scopic view of discourse structure analysis. For instance, a speaker sometimes uses a number of structured UUs to convince his interlocuter to do some particular actions. In such cases, the first UU may presummarize the speaker's proposal, the second UU may talk about his main plan, and the last UU may show the alternative plans. The prosodic information can be also used as a cue for this sort of structure. The structure discussed here is similar to the *argumentative structure* [Cohen 87] or *coherent structure* [Hobbs 79], but slightly different due to our UU based analysis. The rest of this section discusses the relation between prosodic structure and the argumentative structure by showing the typical excerpts from our database.

Fig.5-1 shows a typical (speech act) continuation utterance sequence. Please note that the pitch contour in the figure is simplified by three parameters; onset, first peak, and final pitches. The first peak of each U_i declines toward the end of the sequence, indicating that the topic is not changed. The final pitches other than U_3 's are higher, showing that the relation between the UUs are continuation.

Fig.5-2 shows a simple hybrid case. U_1 and U_2 's prosodic parameters suggest the relation between them is continuation. U_0 's final pitch shows the finality of its proposition and the first peak declination of the UUs indicates that they are concerned with the same topic. Consequently, the prosodic structure suggests the relation between U_0 and U_1 — U_2 block is elaboration class (Elb). Further semantic

analysis may identify it as a pre-summarization.

Final sample shown in Fig.5-3 is more complicated one. U_1 and U_3 are elaborations of preceding utterances— U_0 and U_2 , respectively—and the relation between U_0 , U_2 , and U_4 is continuation. In this case, the discourse structure identification from its prosodic structure is not so simple. U_1 's first peak is drastically suppressed, indicating it is completely subordinate to U_0 . Because of this subordination, U_0 's high final pitch can be taken as continuation to U_2 rather than to U_1 . To clarify the prosodic function concerned with continuation, elaboration, and subordination needs further systematic analysis and we leave it as another future work.

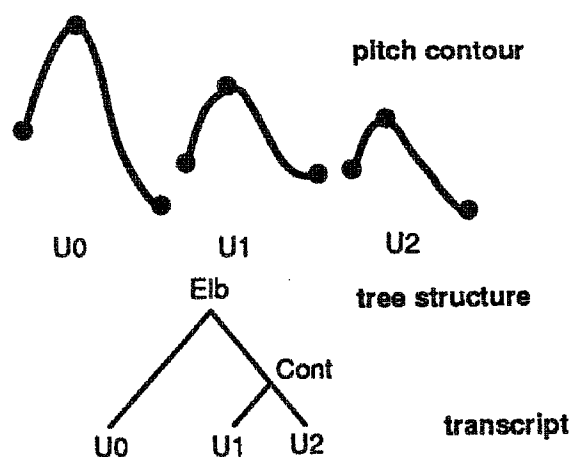
Acknowledgements

Many thanks to Tim Becker for being kindly our subject, and also to David Traum for his fruitful suggestions on the discourse marking.

References

- [AllPer 80] Allen, J.F. & Perrault, C.R. *Analyzing intention in utterances*. Artificial Intelligence 15, 1980.
- [AllSch 91] Allen, J.F. & Schubert, L.K. *The TRAINS project*, TRAINS Technical Note 91-1, Computer Science Dept, University of Rochester, 1991.
- [Austin 62] Austin, J.L. *How to do things with words*. Oxford University Press, 1962.
- [BroYul 83] Brown, G. & Yule, G. *Discourse analysis*. Cambridge University Press, 1983.
- [BroCur 80] Brown, G., Currie, K.L. & Kenworthy, J. *Questions of intonation*. Croom Helm, 1980.
- [Cohen 87] Cohen, Robin. *Analyzing the structure of argumentative discourse*. Computational Linguistics 13, 1987.
- [Guss 83] Gussenhoven, C. *On the grammar and semantics of sentence accents*. Language Sciences 16, 1983.
- [HakSat 80] Hakoda, K. & Sato, H. *Prosodic rules in connected speech synthesis*. Trans. of the Institute of Electronics and Communication Engineers 63-D, 1980.
- [Hobbs 79] Hobbs, J. *Coherence and coreference*. Cognitive Science, 3(1), 1979.

- [Ladd 84] Ladd, D.R. *Declination: a review and some hypotheses*. Phonology Yearbook I, 1984.
- [LiePie 84] Lieberman, M. & Pierrehumbert, J.B. *Intonational invariance under changes in pitch range and length*, in M. Aronoff and R.T. Oehrle (eds.) *Language sound structure*. MIT Press, 1984.
- [NakAll 91] Nakajima, S. & Allen, J.F. *A study of pragmatic roles of prosody in the TRAINS dialogs*. TRAINS technical note, Computer Science Dept, University of Rochester, forthcoming.
- [PieHir 90] Pierrehumbert, J. & Hirschberg, J. *The meaning of intonational contours in the interpretation of discourse*, in P.R. Cohen, J. Morgan, & M.E. Pollack (eds.) *Intentions in communication*. MIT Press, 1990.
- [Searle 69] Searle, J.R. *Speech Acts*. Cambridge University Press, 1969.



H: uhh where.. where are my beer factories
 S: the beer factories are city D and E

u0 S: there also is beer in storage
 u1 S: uhh there's uhh 20 loads, tanker loads of beer at D already
 u2 S: and 55 loads of beer at E already

Fig. 5-2 Hybrid UU sequence example

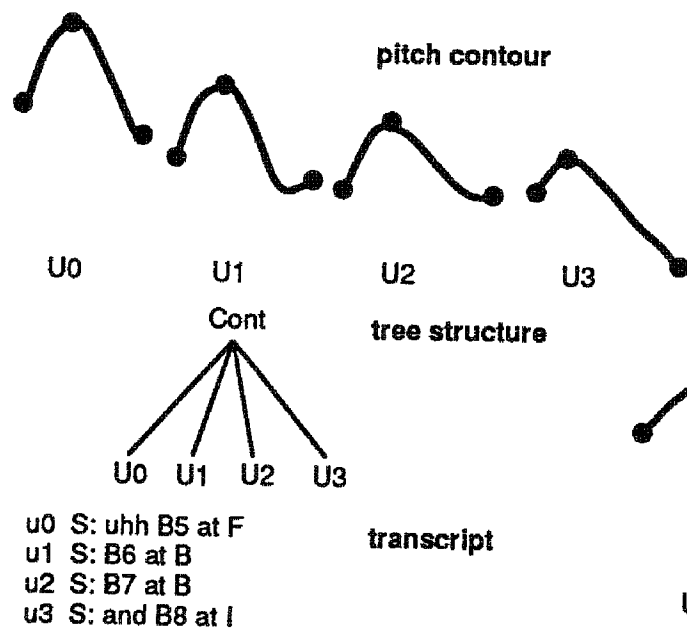
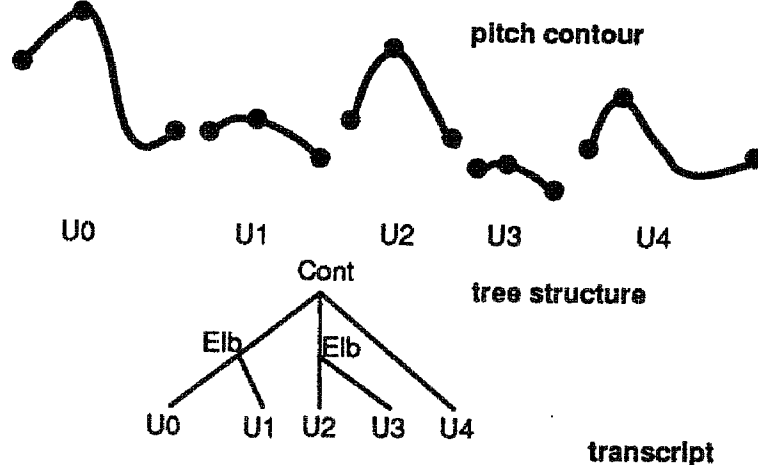


Fig. 5-1 Continuation UU sequence example



u0 H: so let's take..leave tanker T1 at A
 u1 H: we just don't need to take it with us
 S: okay, yes
 u2 H: and we'll use engine E3 and go to city I
 S: okay, let's uh..
 u3 H: by the most direct route
 S: that takes six hours
 u4 H: and there's an orange source there I see

Fig. 5-3 Complicated UU sequence example

発 表 論 文

別 刷 り

音声対話システムにおける 音響処理と言語処理の接点

Combination of Acoustic and Language Processing
in Speech Conversational System

白井 克彦

Katsuhiko Shirai

早稲田大学理工学部情報学科

Department of Information Science,

Waseda University

Abstract: The history of the research on Speech Understanding System (SUS) consists of the efforts continued more than 20 years. That was initiated by the DARPA SUS project 1971. In 80's and 90's, there have been remarkable advance in the technology of continuous speech recognition. One is the improvement in the acoustic domain to discriminate phonemes using HMM or neural networks. The other is the combination of language processing, which includes statistical models and several kinds of grammars. However, there are many problem to be solved in the next stage in order to realize speech conversational systems that can recognize spontaneous speech. Several new directions will be discussed such as the application of prosodic information.

1. はじめに

音声によって表現され、伝達される情報には、文字表記を用いて適切に記述される部分といわゆる韻律情報のようにその記号表記があまり簡単ではなくて、これまでその表記法も十分には整えられていない部分がある。これら2つの部分は、当然、密接に関係しながら用いられているものであるから、不可分とも言えるが、従来、言語情報と感情表現などという呼び方がなされてきた。これまでの音声認識研究は、音声パターンの識別から始まり、言語情報を扱う段階に至って来ている。

1950年代から始まり、1970年代にかけてDPの応用を中心に孤立発声から連続発声された単語認識の研究が大きな進歩をとげたが、ここでは、音声特有の特に音響レベルで見られるパターンの変形が問題となっていた。したがって、言語としての音声処理という段階に至っていなかった。ただし、音声タイプライタという考えは当初から持たれており、音声言語を文字表記に変えるための音韻単位の認識は古くから試みられてきている。

1971年にアメリカ国防総省の援助のもとで行われた音声理解システムの研究は、言語としての音声認識研究を大規模に始めたものとして、エボック・メイキングと言える。その後の音声理解に関する研究は、根本的な発想の転換がなされた事柄は少ないが、パターン・マッチング技術における、HMM法あるいは、さらに広義の統計的な学習方式、ニューラル・ネットワークの適用など、認識精度の著しい向上が起こったことと、言語的情報を実時間処理の枠組の中で組み合わせていくことが可能になりつつあることに大きな進歩が見られる。

最近の進歩の背景には、大規模な音声データ・ベースの存在を見逃すことはできない。従来、音声データ・ベースは各組織毎に集められ、ほとんどそれらは公開利用されることはなかった。しかし、大量に質的に意義のあるデータを集め、整理してデータ・ベース化するには、言うまでもなく多大の費用と労力を要する。米国における、TIやDARPAの後援によって作られた音声データ・ベースは、共通データ利用による研究の進展に大いに効果的であったと言える。日本においても、電子協音声データ・ベース、ATR

音声データベース、文部省科学研究費によるいくつかの音声データ・ベースが作られ共同利用され効果的に利用され始めている。

大規模音声データ・ベースの有効性は、まず、音響レベルにおける音声の多様性に対して、統計的に十分な一般的なモデルを作れるようにした点である。現在のところ、このモデルの枠組の中心になっているのはHMMであるが、将来的にはニューラルネットワークとの組合せなど、より複雑な構造が一般的に用いられるようになるであろう。しかし、その要点は、十分な量の音声データから、その多様性を示す統計的に意味のあるパラメータを確実に取り出すことである。また、パターンの識別問題としては、識別率が最大になるパラメータのチューニングが一般的な枠組として議論されている。

つぎに重要なことは、主として統計的な言語モデルが作られ、連続音声認識に有効に利用されるようになってきたことである。統計的な言語モデルは、現在ではテキスト・データを中心に抽出されており、話し言葉の統計的な言語モデルと呼ぶべきものはあまりない状態でこれから問題である。

連続音声についても、音声データ・ベースの整備にしたがって、米、欧、日で研究が進み、数百語から1千語程度で、明瞭な連続音声について、かなりの高い認識率が得られるに至っている。しかしながら、これらの実験的システムで対象としている音声は、通常の音声対話で現れる発話とは、まだ大きな開きがある。よく指摘されているように、会話音声では、書き言葉のように文構造が整っていないことや多くの省略がみられる。また、より単純に問題を困難にしていることは、多くの部分で発声が速くなったり不明瞭になったりすることが普通なことである。これまでの連続音声認識の立場は、可能な限り音韻や単語単位の認識を行い、統語解析、意味解析とを合わせて、最も確率の高い文章候補を認識結果するものである。

音響量を y として単語列を W とすると、問題は可能な W のなかから、確率最大の W を見つけることである。

$$P(W|y) = \max P(W|y)$$

$$P(W|y) = P(W) P(y|W) / P(y)$$

このシャノン流あるいはベイズ流の問題の定式化が基本的にも有効であるとしても、対話者の思考モデルと音声の生成・受聴の間の関係が入ると、問題が少し変化するように思われる。

音声対話に関する研究は、近年、急速に注目されるようになってきた。これは将来の実践的な音声対話システムの実現のためには、従来の書き言葉をもとに捉えてきた自然言語処理ではなく、話し言葉について研究する必要が一層、明らかになってきたことによるであろう。現在、より自然な状態の音声対話データを収集し、データ・ベース化して研究を強力に進展させたいという動きが生まれつつある。

本稿では、連続音声認識システムにおける言語処理の現状について若干の考察を加え、将来の音声対話システムの構成について考えてみる。

2. 音声理解システム

第1世代の研究は、DARPAプロジェクト(1971~1975)で、内容は、語数1000、人工的構文、タスク限定であった。

代表的システムとしては、

①Hearsay II (CMU)

黑板モデルを生んだ。複数の異なる知識源が非同期、並列的に動作する。

②HWIM (BBN)

階層構造を持つシステム。トップダウンとボトムアップ処理を理想的に追求した。

③HARPY (CMU)

知識表現はすべて、遷移ネットワークに埋め込まれていた。木探索法にビーム・サーチ法を適用。1,011語の情報検索。

日本においても、早大(WABOT I 1972)、NTT(当時電々公社)、京大、京都工繊大、山梨大などで様々なシステムが研究された。要素技術のレベルが未熟で、一般に単語数が少ないが、対話制御などの

考え方も既に含まれていて、現在の音声対話システムの基本構造が示されている。

1980年代には、MIT、IBM、CMUなどを中心に研究が進んだが、どちらかと言えば要素技術に多大な進歩の見られた時代と言える。

1) DP法と言語処理

数字のような連続単語音声認識から進んで、何らかの文法構造を持つ連続単語音声に対してDP法を適用して認識する方法が提案された。それらの中でも最も古いものは、1970年代初頭のVintsyukによるものである。文構造が限られているが、対話の状態依存性を含めて最適化したのは、WABOT Iの音声対話系 Shirai, K., Fujisawa, H. (1974) である。その後、迫江、Bridle、Ney、中川、好田、尾関らの研究が続いた。文法としては、有限オートマトンから文脈自由文法へと進み、好田、尾関の研究は日本語の係り受け解析の最適化を問題としている。これらのアルゴリズムは、言語モデルによってもたらされる様々な形の拘束をDP法で有効に処理できることを示したもので、その後の音声理解システムの中でも基本技術となっている。

たとえば、最近の東北大学のシステムでも、構文駆動型連続DP法が考案されている。この方法はオートマトンで記述された日本語の文節構造を連続DP法を適用して検出するもので、単純なワード・スポッティングよりも有効と思われる。

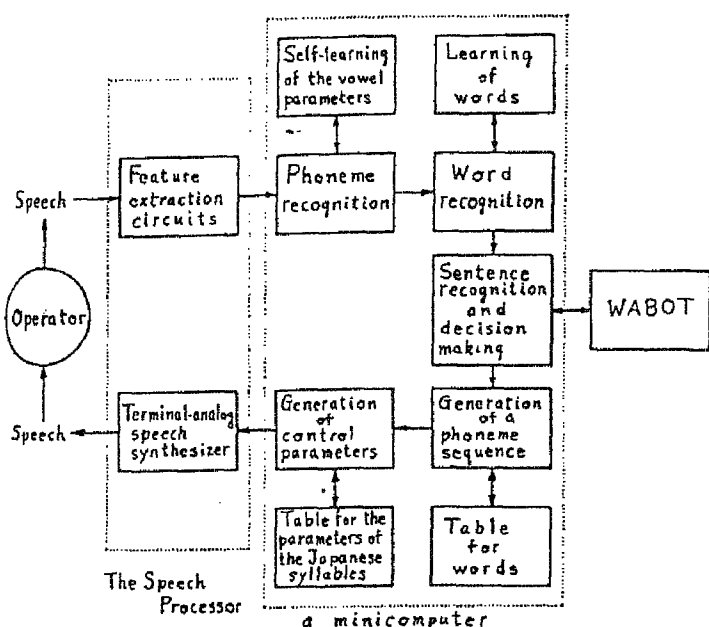


図1. WABOT I のシステム図

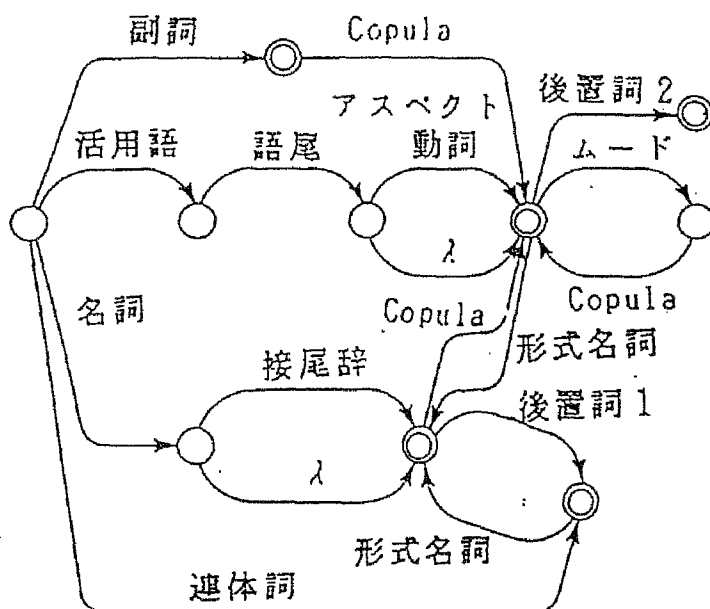


図2. 日本語文節構造のオートマトンモデル（牧野ら）

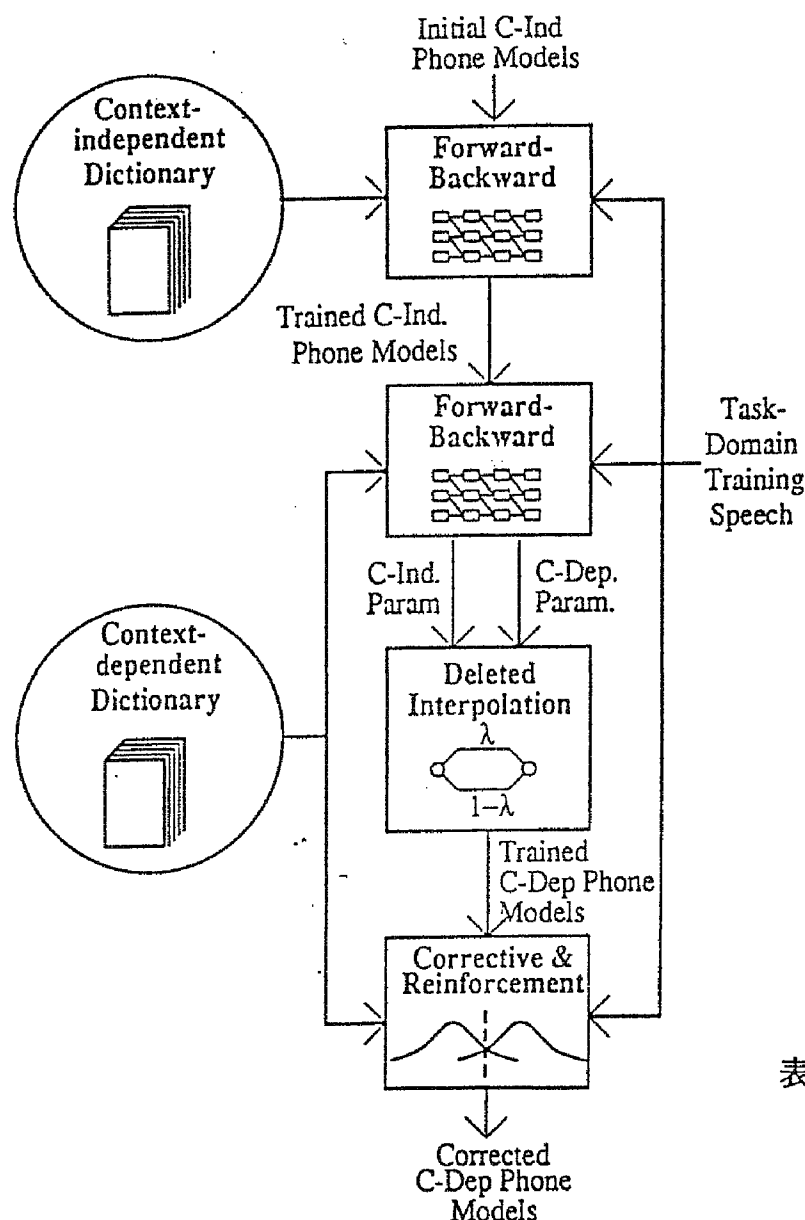
2) HMM法と言語モデル

音声の音響的特性の時間変化に対してマルコフ・モデルを適用することは大変自然なことと思われる。勿論、厳密には人間が発話する際は、単語や文章があるまとまりを作って調音制御、運動生成されるのであるから、マルコフ・モデルは完全ではない。音声認識にマルコフ・モデルを当てはめた例は古くから存在する。たとえば、Shirai, K., Fujisawa, H. (1973) では、単語に対してマルコフ・モデルが作られている。ただし、この場合の状態は擬音韻である。HMMの理論的起源は、音声ではないが、Baker (1975) に始まって、IBM, Bell研を中心に音声認識に適用する研究がなされ、今日、音声認識の音響レベルでのマッチングの基本技術となっている。

言語モデルとしては、CFGが比較的多く用いられているが、HMM法と言語処理を有効に組合せたシステムは、bigramあるいはTrigramと呼ばれる統計的な言語モデルも英語では特に有効に用いられている。

$$\text{言語モデル } P(W) = \prod_{i=1}^n P(W_i | W_1 \dots W_{i-1})$$

$P(W_i | W_1 \dots W_{i-1})$ は単語列 $W_1 \dots W_{i-1}$ の後に W_i が発せられる確率である。



Version	No Grammar	Word Pair
1 Codebook	25.8%	58.1%
3 Codebooks	45.3%	84.4%
+Duration	49.6%	83.8%
+Fn-word	57.0%	87.9%
+Fn-phrase	59.2%	88.4%
+Gen-triphone	72.8%	94.2%
+Between-word	77.9%	95.5%
+Corrective	81.9%	96.2%

表 1. Results of various versions of SPHINX

図 3. The SPHINX Training Procedure (Lee)

$$\text{Trigramモデル } P(W) = \prod_{i=1}^n P(W_i | W_{i-2}, W_{i-1})$$

$P(W_3 | W_1, W_2)$ をデータの中から頻度によって推定することは困難であるから、補完法によって求める。

$$P(W_3 | W_1, W_2) = q_3 f(W_3 | W_1, W_2) + q_2 f(W_3 | W_2) + q_1 f(W_3)$$

ただし、

$$f(W_3 | W_1, W_2) = C(W_1, W_2, W_3) / C(W_1, W_2)$$

C は頻度。 $q_1 + q_2 + q_3 = 1$ 。

HMMと言語処理を結合した試験システムは、各所に見られるようになったがCMUのK.F.Leeらによる Sphinxが代表的である。Sphinxシステムは、DARPAのResource Managementの音声データを用いて研究された。HMM音韻モデルを用い、これを連結して、Viterbiビームサーチによって認識する。ケプストラム係数をベクトル量子化して用いるが、符号帳が1つものが基本となっている。Codebooksは、パワーやケプストラムに対して複数符号帳を設けるもの。Durationは状態継続時間を考慮したもの。Fn Word と Fn phrase は a, the などの言い回しにHMMモデルを作るもの。Gen-triphoneは、前後の音韻環境を取り入れたもの。Between-Wordは単語間のコンテキストを考慮した場合であり、Correctiveは学習データで生ずる誤りについて、HMMモデルを修正する訂正学習法を用いたものである。日本では、ATRのHMM-LR、豊橋技大のSPOJUS-SYNOなどがある。

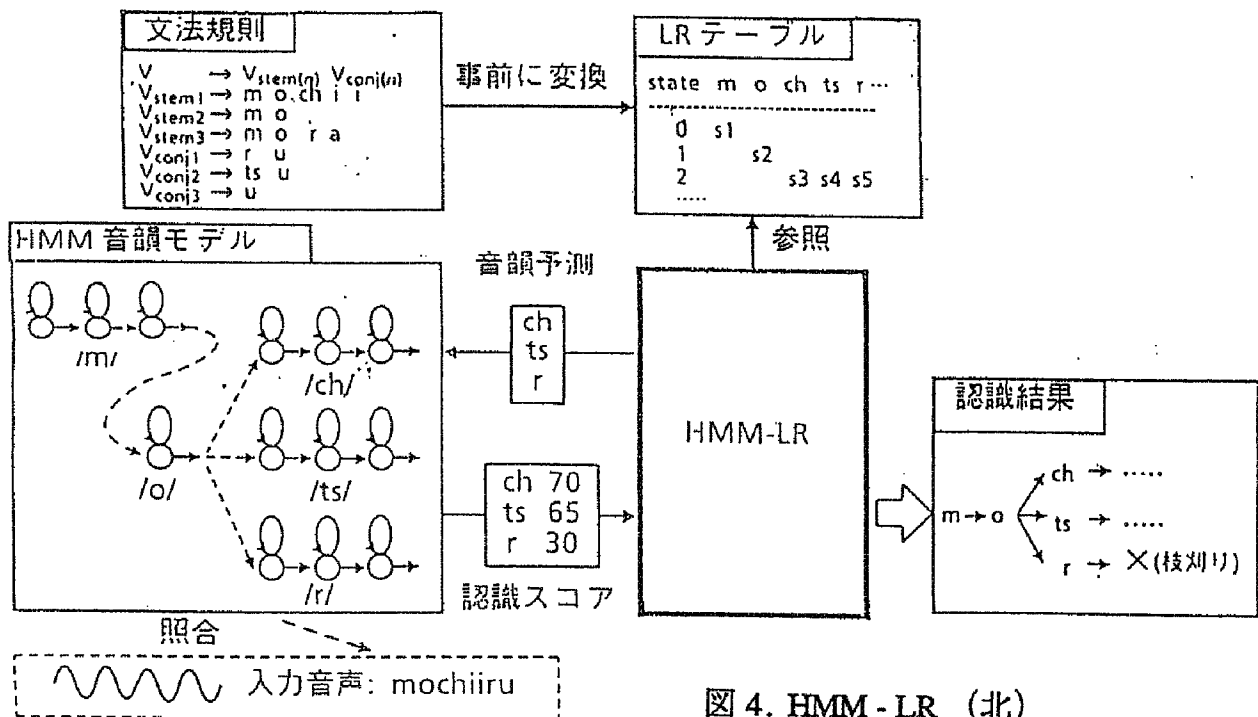


図 4. HMM - LR (北)

3. 音声対話の内容

1) 話者のモデル

話者 (A) or (B) の思考の状態。話者の声の個人性。

2) 話者の話し方、表現。(語彙、文法、韻律)

話者モデルや対話の状態、他の入力に応じた発話の内容。文生成。韻律。相づち。

3) タスクに関する理解、合意

協調的問題解決。プラン・ゴール問題など。

4) 他の入力

雑音などの妨害となるもの。対話に影響を与えるような情報。

5) 対話の主導権

どちらが主導者になっているか。

これまで韻律を認識に組み込んだ研究は大変少ない。韻律における変形は、音韻の変形よりは、はるかに大きいので情報量が少ないとされてきたことと、研究方法もあまり簡単でないことから遅れているが、今後の大きい課題である。

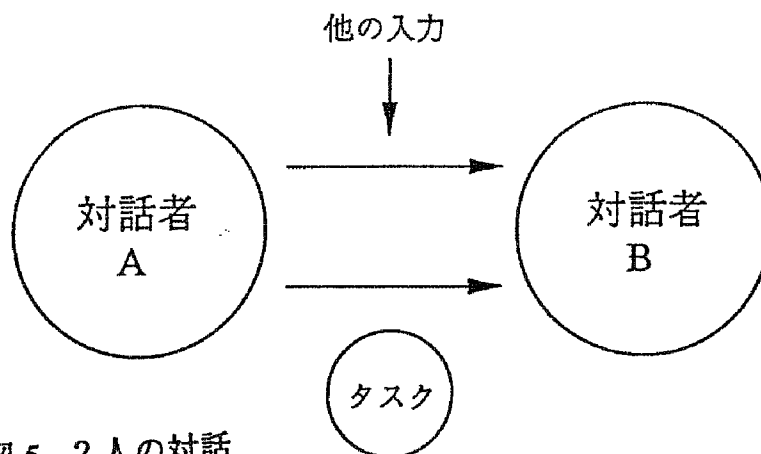


図 5. 2 人の対話

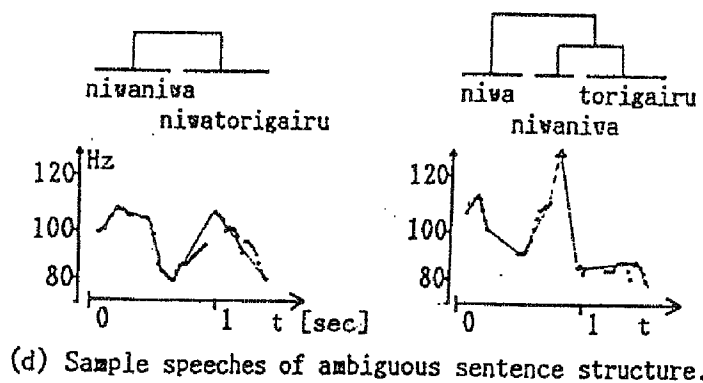
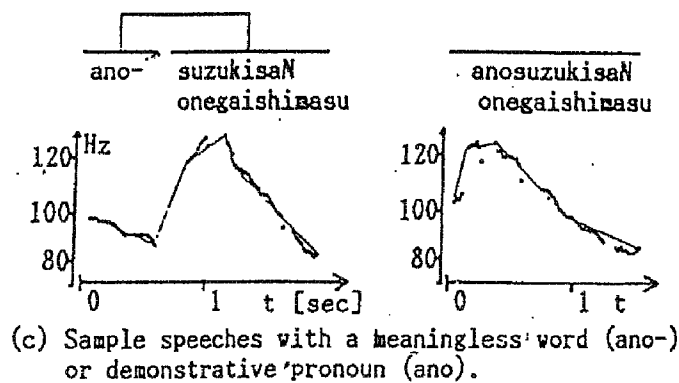
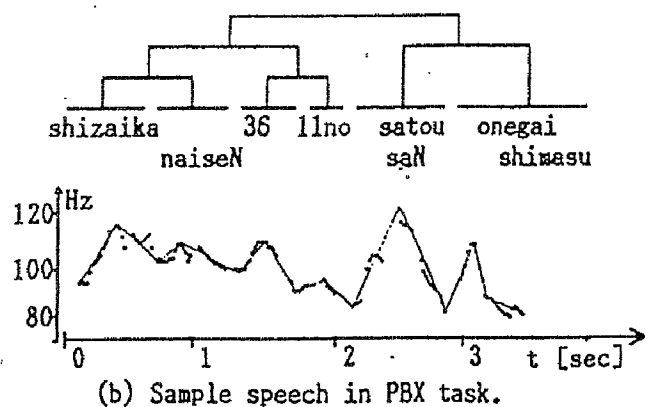
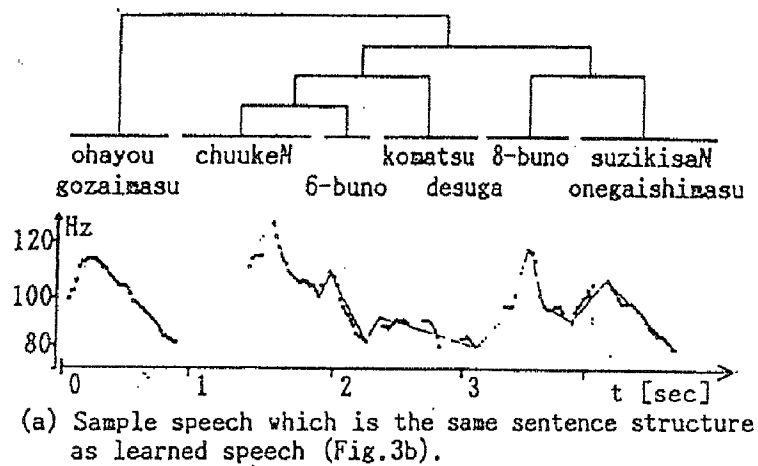


図 6. 韻律情報を利用した構文推定の例 (小松)

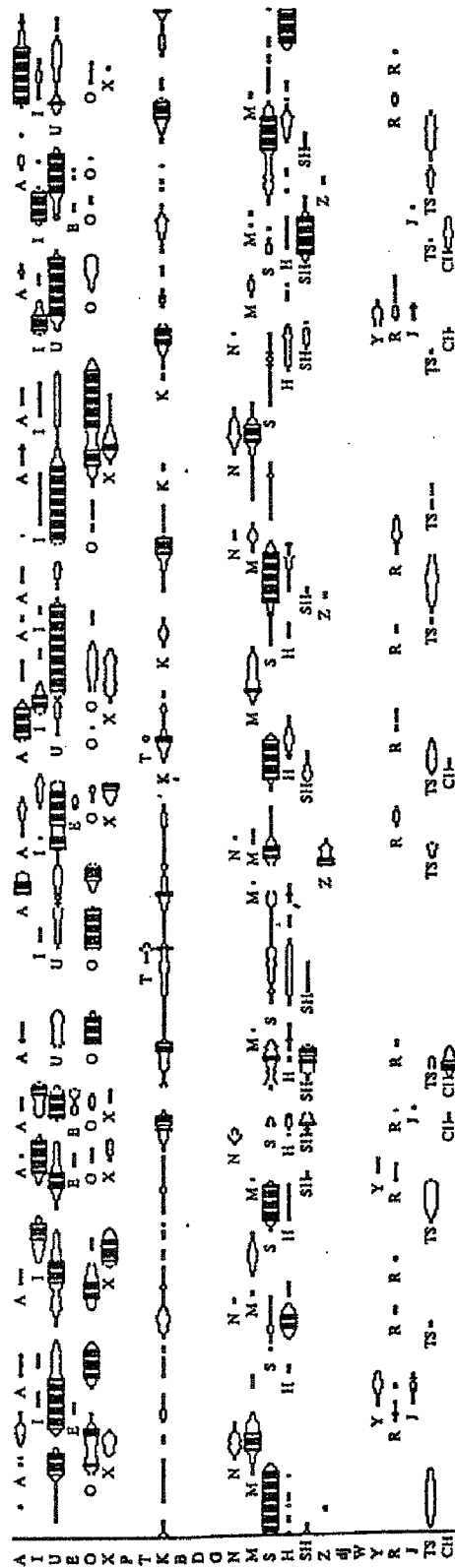
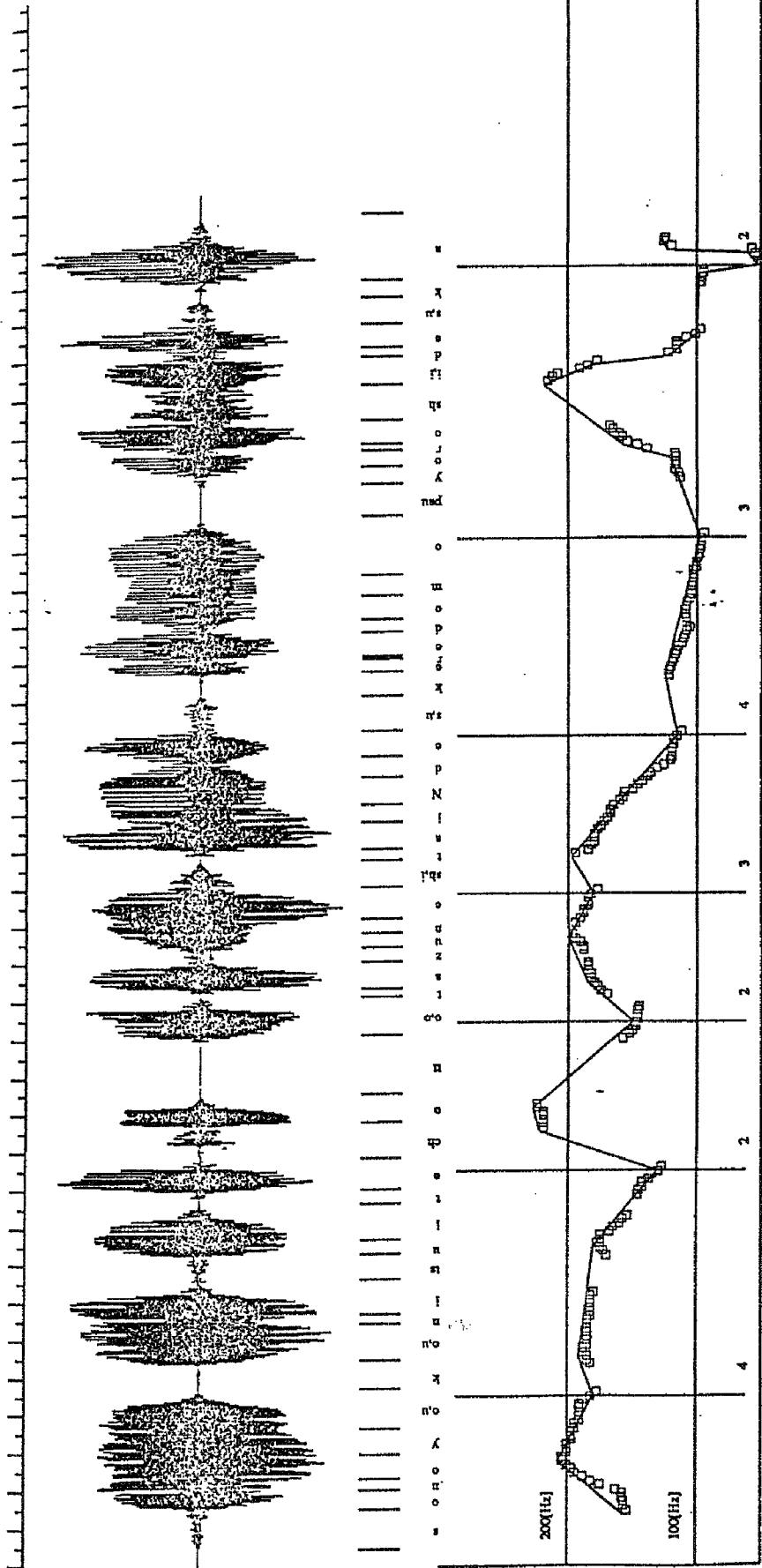


図 8. 音響分析の例 その要項について、ちょっとお尋ねしたいんですけども、よろしいですか

文 献

- [1] Jelinek, F. (1976). Continuous speech recognition by statistical methods. Proc.IEEE 64 (4):532-556.
- [2] Shirai, K., Fujisawa, H. Yukio Sakai. (1973). Ear and Voice of the WABOT. Bulletin of Science and Engineering Reserch Laboratory. No.62, Waseda Univ. (1973), pp.39 - 85.
- [3] Shirai, K., Fujisawa, H. (1974). An Algorithm for Spoken Sentence Recognition and Its Application to the Speech Input - Output System IEEE TRANS Vol. SMC-4, No.5, [1974], pp.475-479.
- [4] Vintsyuk, T.K. (1971). Element-wise recognition of continuous speech consisting of given vocabulary. Kibernetika 2:133 - 143 (in Russian).
- [5] 迫江、藤井、吉田、亘理: "フレーム同期化、ビームサーチ、ベクトル量子化の統合によるDPマッチングの高速化" 信学論誌 D Vol.J71 - D No.9 pp.1650 - 1659.
- [6] Bridle, J., Brown, M.D., and Chamberlain, R.M. (1982). An algorithm for connected word recognition. Proc. ICASSP '82, pp.899 - 902.
- [7] Ney, H. (1987). Dynamic programming speech recognition using a context-free grammar. Proc. ICASSP '87, pp. 69 - 72.
- [8] Nakagawa, S. (1987). Spoken sentence recognition by time-synchronous parsing algorithm of context free-grammar. Proc. ICASSP '87. pp. 829 - 832.
- [9] 松永、好田: "係り受けの整合性に基づく文節認識の後処理", 音学講論集, 1-1-23.
- [10] 尾関: "係り受けの整合度に基づき最適文節列を選択する多段決定アルゴリズム", 信学論, Vol.J70-D, No.3, pp. 601 - 609.
- [11] 岡田、伊藤、牧野、城戸: "構文駆動型連続DP法による連続音声からの活用語のスポッティング", 信学論 D,J70 - D, 12, pp.2479 - 2490.
- [12] Kai-Fu Lee, Hsiao-Wuen Hon, Mei-Yuh Hwang, Xuedong Huang. Speech Recognition Using Hidden Markov Models: a CMU Perspective.
- [13] 中川、大黒、橋本: "構文解析駆動型日本語連続音声認識システム -SPOJUS - SYNO-", 信学論(D - II), J72 - D - II, 8, pp. 472 - 480. (平2 - 3)

機能語予測 CYK 法による日本語文音声の統語処理

正員 伊藤 彰則[†] 正員 牧野 正三[†] 正員 城戸 健一^{†*}

A Functional Word Prediction CYK Method for Parsing Spoken Japanese Sentences

Akinori ITO[†], Shozo MAKINO[†] and Ken'iti KIDO^{†*}, *Members*

あらまし 連続音声認識のための新しい統語処理アルゴリズム「機能語予測 CYK 法」について述べる。機能語予測 CYK 法は、CYK 法をベースとし、これに機能語の予測機能を加えたものである。機能語を予測しながらマッチングすることにより、効率的な処理を行うことができる。次に、この機能語予測 CYK 法にビームサーチを導入したアルゴリズムを提案する。また、機能語を効率良く予測するための正規文法（機能語オートマトン）を導入する。これは、従来の文節処理に用いられてきた有限オートマトンと同じものが使用できるため、文節内文法での各種の制約が利用できる。ビームサーチと機能語オートマトンの導入によって、非終端記号数の増加に伴う記憶容量および計算量の増加を抑えることができる。このビームサーチを用いた機能語予測 CYK 法と、文節検出+統語処理の 2 段階の認識方式との比較実験を行った結果、計算量・精度ともに機能語予測 CYK 法が優れていることがわかった。

1. ま え が き

文章音声の認識を行う場合、認識精度を向上させるために文法的な制約を用いることが多い。初期の文音声認識システムでは制約に用いる文法クラスとして正規文法を用いるシステムが多かったが^{(1),(2)}、正規文法では小規模なタスクしか構築できないため、より大きなタスクを用いるシステムでは文脈自由文法が用いられることが多い。

文脈自由文法を用いて音声認識の統語処理を行うアルゴリズムとしては、横型上昇法である CYK のアルゴリズムに基づくもの⁽³⁾、横型下降法である Earley のアルゴリズムに基づくもの⁽⁴⁾、拡張 LR パーザを使う方法⁽⁵⁾⁻⁽⁷⁾、BUP を使う方法⁽⁸⁾などが提案されている。CYK のアルゴリズムに基づく方法は最も単純であるが、不要な部分木を多く生成し、あまり効率が良くないと言われている⁽⁹⁾。これは、CYK 法が上昇型の統語解析法であるため、入力音声の特定の位置に生起する単語や音素などを予測することができず、その結果として

すべての位置ですべての単語/音素の可能性を仮定して処理を行わなければならないためである。これに対し、Earley のアルゴリズムや拡張 LR パーザでは、単語/音素などの予測を行うことによって探索空間を減らし、効率の良い処理を行っている。しかし、これらの下降型統語解析法では、文脈自由文法の統語規則をもとにして予測を行うため、JPSG や HPSG などのように等価的な統語規則数・非終端記号数が非常に多い文法に適用するのは困難だと考えられる。

一方、日本語文音声の認識においては、統語処理を文節検出と文節間の係り受けの 2 段階で行う方法が従来多くのシステムで採用されている⁽¹⁰⁾⁻⁽¹²⁾。このように処理を 2 段階に分ける利点としては、次のものが挙げられる⁽¹³⁾。

(1) 日本語のように補語や修飾語相互の語順が比較的緩やかな言語では、文脈自由文法などによる統語規則による処理は効率が良くない。文節は一つの意味的単位であり、意味的単位相互間の係り受けによって文を処理した方が、語順の比較的緩やかな日本語の統語処理に適している。

(2) 文節内文法はほぼ正規文法で記述できる⁽¹⁴⁾。そのため、文節の処理のみを考えれば、入力音声と辞書とのマッチングの際に構文制御が容易に行える。

[†] 東北大学応用情報学研究センター，仙台市
Research Center for Applied Information Sciences, Tohoku University, Sendai-shi, 980 Japan
^{*} 現在，千葉工業大学情報工学科

(3) 音素を単位とした認識結果から単語のスポンディングを行う際には、長さの短い単語はマッチング精度が良くない⁽¹⁵⁾。短い単語のほとんどは機能語であるから、これらを実質語に付属させた「文節」を扱うことにより、マッチングの精度が向上する。

しかし、このように処理を2段階に分けることには欠点もある⁽¹⁶⁾。2段階の処理においては、統語処理での情報を文節検出にフィードバックすることができないため、統語処理の段階での予測的な情報は文節検出の段階では利用できない。従って、文節検出の際には、すべての位置にすべての文節の存在を仮定し、網羅的な処理を行わなければならない。また、文節検出の精度を上げようとする、生成される文節ラティスの規模が急激に増大し、文節検出のための計算量が増大すると共に、統語処理の段階での負荷も大きくなってしまふ。

そこで、文節検出と統語処理を同時に行い、処理の各段階において次に現れる可能性のある単語/文節を予測することによって、むだな文節の検出を防ぐことができる。

本論文では、日本語の特性を考慮し、機能語の予測を行うことによって効率を向上させた統語処理アルゴリズム「機能語予測 CYK 法」を提案する。日本語の文法を文脈自由文法で記述する場合、いわゆる「文節内文法」と「文節間文法」にあたる規則を分離することができる。これを利用して、予測が容易な「文節内文法」の部分のみを用いて予測動作を行うことが本アルゴリズムの特徴である。

2.では、機能語予測 CYK 法の考え方と、基本的なアルゴリズムを示す。3.では、実際のシステムへの応用を考慮し、ビームサーチを導入したアルゴリズムを示す。4.では、従来の統語処理法との比較評価を行う。

2. 機能語予測 CYK 法

ここでは、機能語予測 CYK 法の考え方と基本的なアルゴリズムについて述べる。機能語予測 CYK 法では、基本的な CYK のアルゴリズムに加えて、機能語の予測を行う。まずはじめに、文の解析に用いる文法を定義する。

2.1 解析に用いる文法

文の解析に用いる日本語の文脈自由文法を $G=(N, T, P, S)$ とする。但し、 N, T, P, S は、それぞれ非終端記号の集合、終端記号の集合、書換え規則の集合、始端記号である。

- | | |
|---------------------------------|--------------------------------|
| [1] $S \rightarrow TNP VP$ | [12] $VP \rightarrow VPN$ る |
| [2] $NP \rightarrow OVP NP$ | [13] $VP \rightarrow VPN$ た |
| [3] $OVPK \rightarrow SNP TVPK$ | [14] $OVP \rightarrow OVPK$ く |
| [4] $OVPN \rightarrow SNP TVPN$ | [15] $OVP \rightarrow OVPK$ いた |
| [5] $VPK \rightarrow ONP TVPK$ | [16] $OVP \rightarrow OVPN$ る |
| [6] $VPN \rightarrow ONP TVPN$ | [17] $OVP \rightarrow OVPN$ た |
| [7] $SNP \rightarrow NP$ が | [18] $NP \rightarrow$ 彼 |
| [8] $TNP \rightarrow NP$ は | [19] $NP \rightarrow$ 私 |
| [9] $ONP \rightarrow NP$ を | [20] $NP \rightarrow$ 本 |
| [10] $VP \rightarrow VPK$ く | [21] $TVPK \rightarrow$ 書 |
| [11] $VP \rightarrow VPK$ いた | [22] $TVPN \rightarrow$ 借り |

図1 簡単な日本語文法の例
Fig. 1 A simple Japanese grammar.

終端記号の集合 T を、実質語 T_c と機能語 T_f に分割する。 $T_c \cup T_f = T$ かつ $T_c \cap T_f = \emptyset$ である。 T_c は名詞、動詞(語幹)、形容詞(語幹)、形容動詞(語幹)、副詞、連体詞の集合であり、 T_f はそれ以外の単語(あるいは語尾)の集合である(「また」「しかし」などの接続詞は、副詞に含めて考える)。このとき、書換え規則 P は、次の形の規則のみを含むとする。

(Type A) $\alpha \rightarrow \beta\gamma$ ($\alpha, \beta, \gamma \in N$)

(Type B) $\alpha \rightarrow \beta a$ ($\alpha, \beta \in N, a \in T_f$)

(Type C) $\alpha \rightarrow a$ ($\alpha \in N, a \in T_c$)

このような規則の例を図1に示す。図中の規則のうち、[1]から[6]は Type A の規則であり、二つの句の融合を表す。また、[7]から[17]までは Type B の規則であり、句と機能語との接続を表す。[18]から[22]までは Type C であり、語尾規則である。日本語では機能語(付属語・語尾)は独立して現れず、必ず他の要素に後続する形で現れることから、これらの形での日本語の文法が記述できることは容易に理解できるであろう。

2.2 機能語予測 CYK 法

まず、機能語予測 CYK 法の基本的な考え方について述べる。ここで解析に用いる文法 G は、Type A および Type C の規則のみに着目すると、Chomsky 標準形になっていることがわかる。そこで、Type A と Type C の規則については CYK 法を用いて解析し、Type B の規則は別に解析を行う。Type B の規則の右辺の2個目の要素は必ず終端記号であるから、Type B の規則については直接的な単語の予測が可能である。この性質を利用し、Type B の規則を処理する場合には、その後続く可能性のある機能語を予測する。

ここからもわかるように、機能語予測 CYK 法では、機能語についてしか予測を行わず、実質語については予測動作を行わない。これは Earley 法や LR 法に比べて不利に思える。しかし、対象を一般的な日本語に限っ

た場合、少なくとも文法的には、実質語の予測は困難である。例えば、一般的な日本語の文章では、「私は」という句の後には

私は 彼女を(連れていく) [名詞]
 私は いく [動詞]
 私は 早い(時間に到着した) [形容詞]
 私は いつも(会社まで歩く) [副詞]

など、すべての実質語がくる可能性がある。そのため、Earley 法や LR 法によって単語の予測をしたとしても、実際に予測が可能なのは機能語に限られることになる。一方、文節として完成していない句、例えば、

歩い(た/て/…)

のような句の場合には、「その後に実質語が接続しない」という形で予測が可能である。しかし、連続音声認識に応用することを考えた場合、「ある位置で終端する句」として複数の可能性をもたせるのが普通である。そのとき、ある位置で終端する複数の句の中に、文節として完成している句が一つでもあれば、その位置から後ろにはすべての実質語が後続する可能性がある。実際の解析においては、そのような句が一つもない位置はほとんどなく、ほとんどすべての位置ですべての実質語をチェックせざるを得ない。従って、単語の予測機能をもつ手法と機能語予測 CYK 法を比較しても、予測効率に大きな差はないものと考えられる。

2.3 アルゴリズム

2.1 で定義した文法 G を用いた、機能語予測 CYK 法のアルゴリズムを示す。入力としては、連続発声された文音声から音素認識を行って得られた音素系列、あるいはフレームごとに入力パターンと音素標準パターンをマッチングした結果などを想定している。マッチング結果はゆう度で表され、これを最大化する。

まずはじめに、アルゴリズム記述のための記号を定義する。解析のための文法を $G=(N, T, P, S)$ とし、Type A, B, C の書換え規則の集合をそれぞれ P_A, P_B, P_C とする。また、Type A, B, C の規則の左辺に現れる非終端記号の集合をそれぞれ N_A, N_B, N_C とする。 ϵ は空列、 ω は N に含まれない記号とする(ω は非終端記号に対応しない部分木を表すために使用する)。文法 G に属する非終端記号 α と単語列 w 、およびゆう度 l の組

$\langle \alpha, w, l \rangle, \alpha \in N \cup \{w\}, w \in T^*$

を部分木と呼ぶ。また、部分木 $t_1 = \langle \alpha_1, w_1, l_1 \rangle$ と $t_2 = \langle \alpha_2, w_2, l_2 \rangle$ を規則 $\gamma \rightarrow \alpha_1 \alpha_2$ によって融合した部分木を $U(t_1, t_2, \gamma \rightarrow \alpha_1 \alpha_2) \equiv \langle \gamma, w_1 w_2, l_1 + l_2 \rangle$ と定義する。更に、

$\lambda(\langle \alpha, w, l \rangle) \equiv l$ とする。その他に、次の記号を定義する。

I	入力系列長
$X_{j,i,c}$	始端 j , 終端 i で、非終端記号 c に対応する部分木 ($1 \leq i \leq j \leq I$)
$Sc(j, i w)$	入力系列 j から i までと終端記号 w とのマッチングゆう度
θ	マッチングのしきい値
$PC(i)$	位置 i で終端する部分木の非終端記号の集合
PR	ある位置で予測される Type B の規則の集合

以上の定義のもとに、機能語予測 CYK 法のアルゴリズムを図 2 に示す。機能語予測 CYK 法の処理は、大きく (1) 実質語のマッチング、(2) 部分木の融合、(3) 機能語の予測、(4) 機能語のマッチングに分かれる。(1) と (2) の部分は CYK 法での処理とほぼ同じであり、Backward time alignment⁽³⁾ による実質語のマッチングと、書換え規則の適用を行う。これらの処理の様子を図 3 に示す。(1) では、ある位置 i を終端として、すべての実質語を始端フリーでマッチングする (Type C の規則の適用)。(2) では、 i を終端とするすべての区間 ($j \sim i$) に注目し、その区間を二つに分け、前半にある部分木と後半にある部分木の間で Type A の規則を適用する。ここまでの処理が終わると、位置 i で終端するすべての部分木が求まる。(3) では、位置 i で終端するすべての部分木を調べ、それらに対して適用可能な Type B の規則を選び出す。(4) では、(3) で選ばれた規則について、機能語を予測してマッチングすると共に、その規則を適用して新たな部分木を生成する。図 1 の文法を用いて、(3) および (4) の処理を行っている例を図 4 に示す。図では、既に解析済みの二つの部分木「彼」「本を借り」から五つの機能語「は」「が」「を」「る」「た」が予測されている。それぞれの部分木の左のシンボルはその部分木に対応する非終端記号を示し、またそれぞれの機能語の右の番号は、予測のときに適用された規則の番号を示している。例えば「が」は、7 番目の規則

$SNP \rightarrow NP$ が

によって予測されている。予測されたこれらの機能語は、 $i+1$ の位置から終端フリーでマッチングされ、解析済みの部分木と融合して新たな部分木となる。

このアルゴリズムの計算量は、単語の平均長さを L 、平均予測機能語数を V_p とすると、

```

for all  $1 \leq i \leq I, c \in N$  do
   $X_{i,c} \leftarrow \langle \omega, c, -\infty \rangle$ 
for all  $1 \leq i \leq I$  do
   $PC(i) \leftarrow \phi$ 
  for  $i-1$  to  $I$  do begin
    (実質語のマッチング)
    for all  $\alpha \rightarrow w \in P, \alpha \in N_C, w \in T_c$  do begin
      単語 (終端記号)  $w$  を、終端を  $i$  に固定して始端フ
      リーでマッチングする。この結果、 $Sc(j, i|w), 1 \leq j \leq i$  が
      求まる。
      for  $j=i$  downto  $1$  do
        if  $Sc(j, i|w) > (i-j+1)\theta$  and  $Sc(j, i|w) > \lambda(X_{j,i,\alpha})$  then begin
           $X_{j,i,\alpha} \leftarrow \langle \alpha, w, Sc(j, i|w) \rangle$ 
           $PC(i)$  に  $\alpha$  を加える
        end
      end
    end
    (部分木の融合)
    for  $j=i-1$  downto  $1$  do begin
      for  $k=j$  to  $i-1$  do begin
        for all  $\gamma \rightarrow \alpha\beta \in P, \gamma \in N_A, \alpha, \beta \in N$  do begin
          if  $\lambda(U(X_{j,k,\alpha}, X_{k+1,i,\beta}, \gamma \rightarrow \alpha\beta)) > (i-j+1)\theta$  and
              $\lambda(U(X_{j,k,\alpha}, X_{k+1,i,\beta}, \gamma \rightarrow \alpha\beta)) > \lambda(X_{j,i,\gamma})$  then begin
             $X_{j,i,\gamma} \leftarrow U(X_{j,k,\alpha}, X_{k+1,i,\beta}, \gamma \rightarrow \alpha\beta)$ 
             $PC(i)$  に  $\gamma$  を加える
          end
        end
      end
    end
  end
  (機能語の予測)
   $PR \leftarrow \phi$ 
  for all  $\alpha \in PC(i)$  do
    for all  $\gamma \rightarrow \alpha f \in P, \gamma \in N_B, f \in T_f$  do
       $\gamma \rightarrow \alpha f$  を  $PR$  に加える
    (機能語のマッチング)
    for all  $\gamma \rightarrow \alpha f \in PR$  do begin
      単語  $f$  を、 $i+1$  を始端として終端フリーでマッチング
      する。その結果、 $Sc(i+1, m|f), i+1 \leq m \leq I$  が求まる。
      for  $p=i+1$  to  $I$  do
        for  $m=i+1$  to  $I$  do
           $X \leftarrow U(X_{p,i,\alpha}, \langle \omega, f, Sc(i+1, m|f) \rangle, \gamma \rightarrow \alpha f)$ 
          if  $\lambda(X) > (m-p+1)\theta$  and  $\lambda(X) > \lambda(X_{p,m,\gamma})$  then begin
             $X_{p,m,\gamma} \leftarrow X$ 
             $\gamma$  を  $PC(m)$  に加える
          end
        end
      end
    end
  end
  以上の解析が終わった時点で、 $X_{1,I,\omega} = \langle \omega, \omega, l \rangle$  とする
  と、 $\omega$  が認識結果、 $l$  が認識スコアになる。

```

図2 機能語予測 CYK 法のアルゴリズム

Fig. 2 The algorithm of the functional word prediction CYK method to obtain an optimum result.

$$O((|N_A| + V_p)I^3 + (|N_C| + V_p)IL^2)$$

で表される。

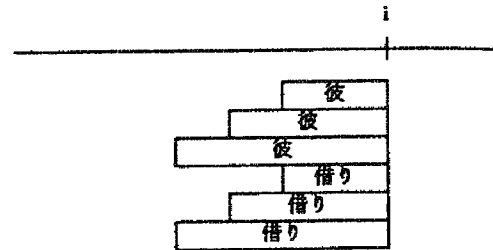
3. ビームサーチの導入

本節では、比較的大きいタスクの統語処理を行う場合の問題点と、それを解決するためにビームサーチを導入した機能語予測 CYK 法のアルゴリズムについて述べる。

3.1 大きいタスクを扱う場合の問題点

文脈自由文法を用いて、ある程度一般性のあるタスクを想定した音声認識を行う場合に問題になるのは、文法記述の問題であろう。単純な文脈自由文法を用いてこのようなタスクを記述すると、書換え規則や非終

【実質語のマッチング】



【部分木の融合】



【機能語のマッチングと融合】

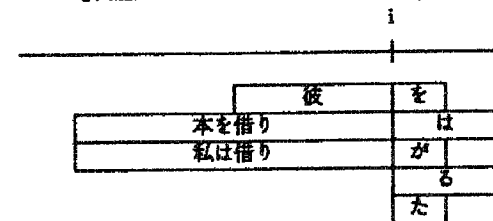


図3 機能語予測 CYK 法による処理

Fig. 3 Parsing based on the functional word prediction CYK method.

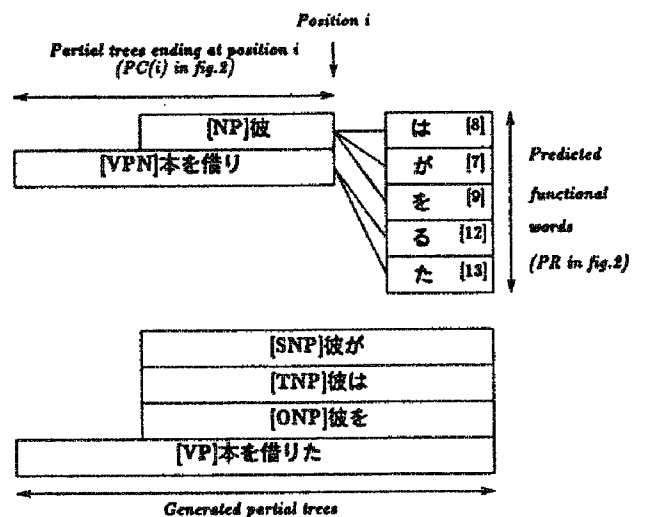


図4 機能語予測の例

Fig. 4 An example of functional word prediction.

端記号の数が膨大になり、文法の構築やメンテナンスの点から見ても望ましいものではない⁽¹⁷⁾。この問題を解決するため、GPSG⁽¹⁸⁾やJPSG⁽¹⁹⁾などのユニフィケーションに基づく句構造文法が提案されてきた。これらの文法では、非終端記号に構造をもたせると共に、少数の原理を与えることで多数の構文規則を間接的に指

彼	{POS N; SUBCAT{}}
私	{POS N; SUBCAT{}}
本	{POS N; SUBCAT{}}
書	{POS V; ITYPE K1; SUBCAT{PP[SB],PP[OBJ]}}
借り	{POS V; ITYPE 2; SUBCAT{PP[SB],PP[OBJ]}}
は	{POS P; ADJUNCT V[SLASH PP]; SUBCAT{N}}
が	{POS P; GR SBJ; SUBCAT{NP}}
を	{POS P; GR OBJ; SUBCAT{NP}}
く	{POS V; ADJUNCT NP; SUBCAT{VP[ITYPE K1]}}
いた	{POS V; ADJUNCT NP; SUBCAT{VP[ITYPE K1]}}
る	{POS V; ADJUNCT NP; SUBCAT{VP[ITYPE 2]}}
た	{POS V; ADJUNCT NP; SUBCAT{VP[ITYPE 2]}}

図5 図1の文法の JPSG による記述例

Fig. 5 An example of the simple Japanese grammar described in JPSG.

定する。これは、Chomsky 標準形の文法 $G=(N, T, P, S)$ において、

$$\alpha \rightarrow a \quad \alpha \in N, a \in T$$

の形の規則を語規則として指定し、また二つの非終端記号 β, γ が与えられたときに、

$$\alpha \rightarrow \beta\gamma \quad \alpha, \beta, \gamma \in N$$

の形の規則が P に含まれるかどうかと、含まれる場合の α の構造を各種の原理によって記述していると考えられる。図1の文法を JPSG で記述した例を図5に示す。各辞書項目に対する素性名と素性値のリスト(例えば「私」に対する {POS N; SUBCAT{ }} は、通常の CFG における非終端記号に対応する。このような枠組みでは、非終端記号が最大(素性の種類×素性値の数)だけ存在するため、等価的な非終端記号数が非常に多くなる。そのため、LR Parser のように予測のための表を作る解析アルゴリズムでは、表のための記憶領域が膨大なものになる。また、Earley のアルゴリズムに代表される下降型統語解析法では、ある時点で適用すべき構文規則を選ぶとき、ある $\beta \in N$ から $\alpha \rightarrow \beta\gamma \in P$ の形の規則を選び出す必要がある。今述べたような文法の枠組みでこの動作を行おうとすると、ある β から導かれる規則 $\alpha \rightarrow \beta\gamma \in P$ が非常に多くなるため、Earley 法のような下降型の統語解析法を用いるのは極めて難しい。

通常、自然言語処理では構造をもったこれらの非終端記号を直接用いず、ふつうの文脈自由規則を使って構文解析を行いつつ機能構造を生成する場合が多い。これは例えば、JPSG における POS 素性のように従来の非終端記号に対応する素性値を用いて文法を構築し、Earley 法などを用いて解析を行う方法である。この方法では、実際に適用される文法のほかに、構文解析用の文法がもう一つ必要になる。これは冗長であると同時に、統語処理の段階でもとの文法のもつ制約を十分

に利用できない可能性がある。

機能語予測 CYK 法では、 $\alpha \rightarrow \beta\gamma \in P(\alpha, \beta, \gamma \in N)$ の形の規則の適用に関しては予測を行わないので、この形の規則については規則の選択および予測表の記憶領域の問題は生じない。しかし、解析の際に非終端記号数に比例する大きさの表 $X_{i,j,a}$ を使用するため、解析表の記憶領域が大きくなり、計算量も多くなるという問題がある。そこで、ビームサーチの導入によってこの問題を解決する。ビームサーチを導入することによって解の最適性は失われるが、記憶領域と計算量を大幅に削減することができる。また、Type B の規則 ($\alpha \rightarrow \beta a, \alpha, \beta \in N, a \in T$) については Earley 法などと同じく規則の選択の必要があるが、機能語オートマトンを導入することによってこの問題を回避する。

3.2 ビームサーチの導入

2.で述べたアルゴリズムでは、入力系列のすべての部分について、すべての非終端記号に対応する部分木の中で最適なものを一つ保存していた。しかし、非終端記号が非常に多い場合、解析表が巨大になると共に計算量も多くなってしまふ。そのため、2.のアルゴリズムでの解析表の代わりに、入力系列のすべての部分でゆが度が高い順に一定個数の部分木のみを保存することにする。このとき、それぞれの部分木がどの非終端記号に対応するかにかかわらず、部分木のゆが度のみで保存する部分木を決定するため、解の最適性は失われる。しかし、保存する部分木数(ビーム幅)を増やしていけば認識率は飽和し、最適解に近くなる。

3.3 機能語オートマトンの導入

2.で定義したアルゴリズムでは、Type B の規則 $\alpha \rightarrow \beta a$ によって機能語の予測を行った。しかし、規則の数が多くなると、ある β についてすべての $\alpha \rightarrow \beta a$ をチェックすることが難しくなってくる。そこで、次のような性質をもつ比較的小規模な正規文法 $G_f=(N_f, T_f, P_f, S_f)$ を使って機能語の予測を行う(これは有限オートマトンと等価である。この正規文法を有限オートマトンとみなした場合、オートマトンの状態遷移図におけるノードは正規文法における非終端記号に対応し、またアークは正規文法の書換え規則に対応する。以下、この正規文法を有限オートマトンとみなし、機能語オートマトンと呼ぶ)。

(性質1) すべての Type C の規則 $\alpha \rightarrow a \in P$ について、 $\phi(a) = cat(a), S_f \rightarrow \phi(a) \in P_f$ である。但し、 $cat(a)$ は $a \in T_c$ の品詞を表し、 $cat(a) \in N_f$ である。また、 ϕ は N から N_f への写像であり、 G の非終端記号と G_f の非

終端記号との対応を表す。

(性質2) すべての Type B の規則 $\alpha \rightarrow \beta a \in P$ について、 $\phi(a) \in N_f$ である。また、 α を右辺にもつ Type B の規則が P に含まれていれば、 $\phi(\beta) \in N_f$, $\phi(\beta) \rightarrow a\phi(\alpha) \in P_f$, そうでなければ $\phi(\beta) \rightarrow a \in P_f$ である。

(性質3) すべての Type A の規則 $\alpha \rightarrow \beta \gamma \in P$ について、 $\phi(\alpha) = \phi(\gamma)$ である。

これらの性質は、この正規文法によって行われる機能語の部分的な解析が、もとの文法による機能語の解析と一致することを表している。性質1は、Type C の規則が、機能語オートマトンにおける実質語アーク (例えば $S_f \rightarrow \text{Noun}'$ のような規則) と対応することを表す。

性質2によって、もとの文法における

$ONP \rightarrow NP$ を

といった規則が、機能語オートマトンにおける

$NP' \rightarrow$ を ONP' あるいは

$NP' \rightarrow$ を

といった規則と対応づけられる。性質3は、Type A の規則の適用によってもとの文法と機能語オートマトンとの対応が崩れないことを保証するものである。但し、この機能語予測のための正規文法を使って機能語の予測をするためには、もとの文法 G に次の制限が必要である。

(制限1) $\alpha \rightarrow \beta a \in P_\beta$ について、 $\beta \in N_A$ ならば、すべての $\beta \rightarrow \gamma \delta \in P_A$ について $\alpha \rightarrow \delta a \in P_\beta$ である。

(制限2) $\alpha \rightarrow \beta a \in P_\beta$ について、 $\gamma \rightarrow \delta \beta \in P_A$ ならば、 $\alpha \rightarrow \gamma a \in P_\beta$ である。

制限1は、ある機能語が二つ以上の実質語を含む句に接続し得るとき、その機能語はその句の最後の要素に接続し得ることを保証する。例えば、「音響学の研究」という名詞句に「には」という機能語が接続可能であるというとき、もとの名詞句の後ろの要素「研究」にも「には」が接続可能であるということである。また、制限2は、ある機能語がある句に接続可能であるというとき、その句が何かに修飾されてできた句に対してもその機能語が接続可能であることを保証する。例えば、「研究」という名詞に「には」という機能語が接続可能であるというとき、その名詞が修飾された「音響学の研究」という名詞句に対しても「には」が接続可能であるということである。日本語の文法が文節間文法と文節内文法に分けて記述できることから、この二つの制限は一般的な日本語の文法についても十分に成り立つと言ってよい。

機能語オートマトンをもとの文法から構成すること

も可能であるが、これは文節構造を表すオートマトン^{(10),(14),(20)}と同じもので、これを利用した方が効率が良く、また各種の精密化⁽²⁰⁾も利用できる(4.の実験では、文献(20)の文節オートマトンを利用した)。

3.4 ビームサーチを用いた機能語予測 CYK 法のアルゴリズム

2.2 で定義した機能語予測 CYK 法にビームサーチを導入したアルゴリズムを記述する。まずはじめに、以下のアルゴリズム記述に用いられる記号を定義する。なお、特に指定のない場合は2.2で定義した記号をそのまま用いる。

以下の記述では、非終端記号 α 、機能語オートマトンのノード(非終端記号) α_f 、単語列 w 、ゆう度 l の四つ組

$\langle \alpha, \alpha_f, w, l \rangle$, $\alpha \in N \cup \{\omega\}$, $\alpha_f \in N_f \cup \{\omega\}$, $w \in T^*$

を部分木と呼ぶ。また、部分木 $t_1 = \langle \alpha_1, \beta_1, w_1, l_1 \rangle$ と $t_2 = \langle \alpha_2, \beta_2, w_2, l_2 \rangle$ を規則 $\gamma \rightarrow \alpha_1 \alpha_2$ で融合した部分木を $U(t_1, t_2) \equiv \langle \gamma, \beta_2, w_1 w_2, l_1 + l_2 \rangle$ と定義する。その他に、以下の記号を定義する。

D 部分木のビーム幅

$X_{i,j,d}$ 始端 i 、終端 j で、 d 番目のゆう度の部分木 ($1 \leq i \leq j \leq I, 1 \leq d \leq D$)

また、次の動作を $put_tree(i, j, t)$ と略記する。入力系列の i から j の位置にある部分木の集合 $S_{i,j} = \{X_{i,j,d} \mid 1 \leq d \leq D\}$ に部分木 t を加えた集合について、その $D+1$ 個の要素をゆう度順にソートし、上位 D 個を新たに $X_{i,j,1} \sim X_{i,j,D}$ とする。但し、 $t = \langle \alpha, \beta, w, l \rangle$ として、 $S_{i,j}$ に $t' = \langle \alpha, \beta, w, l' \rangle$ なる部分木が含まれていた場合は、ゆう度の大きい方を残し、小さい方は $S_{i,j}$ から除く。

以上の定義のもとに、ビームサーチを導入した機能語予測 CYK 法のアルゴリズムを図6に示す。このアルゴリズムと図2のアルゴリズムの主な違いをまとめると、次のようになる。

(1) 図2のアルゴリズムでは、入力系列のすべての位置について、すべての非終端記号について最大のスコアをもつ部分木を1個保存していた($X_{i,j,a}$)。図6のアルゴリズムでは、入力系列のすべての位置について、部分木のカテゴリー(対応する非終端記号)に関係なく D 個の部分木を残す($X_{i,j,d}$)。そのため、ある位置に部分木を登録するときに、部分木のソート(put_tree)が必要である。

(2) 機能語の予測を行う際に、図2のアルゴリズムでは、ある位置で終端する部分木のカテゴリー($PC(i)$)

```

for all  $1 \leq i, j \leq I, 1 \leq d \leq D$  do
   $X_{i,j,d} \leftarrow \langle \omega, \omega, \epsilon, -\infty \rangle$ 
for  $i \leftarrow 1$  to  $I$  do begin
  (実質語のマッチング)
  for all  $\alpha \rightarrow w \in P, \alpha \in N_C, w \in T_s$  do begin
    単語 (終端記号)  $w$  を、終端を  $i$  に固定して始端フリーでマッピングする。この結果、 $Sc(j,i|w), 1 \leq j \leq i$  が求まる。
    for  $j \leftarrow i$  downto  $1$  do
      if  $Sc(j,i|w) > (i-j+1)\theta$  then
        put_tree( $j, i, \langle \alpha, cat(w), w, Sc(j,i|w) \rangle$ )
    end
  (部分木の融合)
   $PR \leftarrow \phi$ 
  for  $j \leftarrow i-1$  downto  $1$  do begin
    for  $k \leftarrow j$  to  $i-1$  do
      for  $d \leftarrow 1$  to  $D$  do
        for  $c \leftarrow 1$  to  $D$  do begin
           $X_{j,k,c} \leftarrow \langle \alpha_1, \beta_1, w_1, l_1 \rangle$ ,
           $X_{k+1,i,d} \leftarrow \langle \alpha_2, \beta_2, w_2, l_2 \rangle$  とする
          for all  $\gamma \rightarrow \alpha_1 \alpha_2 \in P$  do
            if  $\lambda(U(X_{j,k,c}, X_{k+1,i,d})) > (i-j+1)\theta$  then
              put_tree( $j, i, U(X_{j,k,c}, X_{k+1,i,d})$ )
          end
        end
      for  $d \leftarrow 1$  to  $D$  do begin
         $X_{j,i,d} \leftarrow \langle \alpha, \gamma, w, l \rangle$  について、 $\gamma$  を  $PR$  に加える。
      end
    end
  end
  (機能語の予測とマッピング)
  for all  $\gamma \rightarrow f \in P_f, \gamma \in PR, \alpha \in N_f \setminus \{ \epsilon \}$  do begin
    単語  $f$  を、 $i+1$  を始端として終端フリーでマッピングする。その結果、 $Sc(i+1,m|f), i+1 \leq m \leq I$  が求まる。
    for  $j \leftarrow 1$  to  $i$  do
      for all  $\langle \beta, \gamma, w, l \rangle \in \{ X_{j,i,d} | 1 \leq d \leq D \}$  do
        for  $m \leftarrow i+1$  to  $I$  do begin
           $X \leftarrow U(\langle \beta, \gamma, w, l \rangle, \langle \omega, \alpha, f, Sc(i+1,m|f) \rangle)$ 
          if  $\lambda(X) > (m-j+1)\theta$  then
            put_tree( $j, m, X$ )
          end
        end
      end
    end
  end
  以上の解析が終わった時点で、 $X_{1,I,1} \leftarrow \langle \alpha, \beta, w, l \rangle$  とすると、 $w$  が認識結果、 $l$  が認識スコアになる。

```

図6 ビームサーチを導入した機能語予測 CYK 法
Fig. 6 The algorithm of the functional word prediction CYK method with the beam search technique.

から Type B の規則を選び出し、その規則を用いて機能語の予測を行った。図6のアルゴリズムでは、ある位置で終端する部分木の、機能語オートマトンでのノード (PR : 正規文法でのカテゴリーに相当) から、機能語オートマトンを用いて機能語の予測を行っている。

このアルゴリズムの計算量は、単語の平均長さを L 、平均予測機能語数を V_p とすると、

$$O((D + V_p)DI^3 + (|N_c| + V_p)IL^2)$$

で表される。

4. 比較実験

本アルゴリズムと、文節検出+統語処理の2段階の処理とを比較するため、評価実験を行った。

4.1 実験条件

実験に用いたテキストは音素認識率 85 % の擬似音素

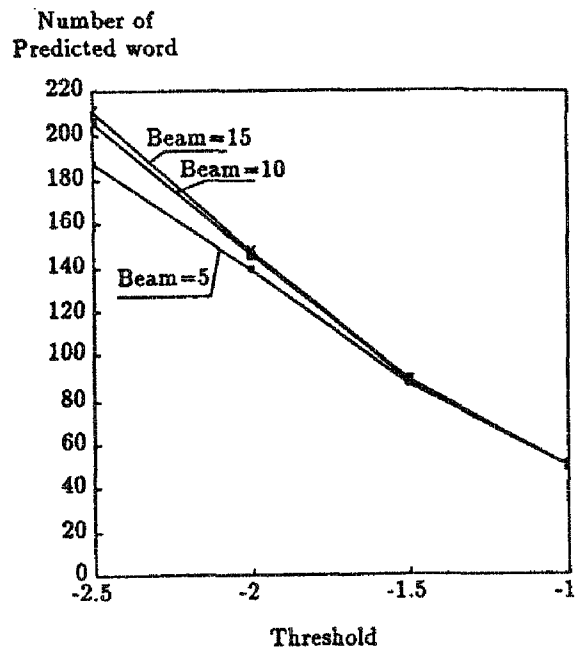


図7 予測される機能語数
Fig. 7 Number of predicted functional words.

系列⁽²¹⁾ 30 文、実質語辞書は 843 単語である。機能語オートマトンは、文献(20)のものを利用した。このオートマトンに含まれる機能語は 431 語、非終端記号数は 106、規則数は 559 である。

4.2 機能語予測の効果

最初に、統語処理の各段階で予測される機能語数について調べた。処理のしきい値 θ とビーム幅 D を変化させたときの平均予測機能語数のグラフを図7に示す。このグラフより、 $\theta = -2.0$ の場合、予測される機能語の数は全機能語数の 30 % 程度になることがわかる。また、予測される機能語数はビーム幅にはあまり関係なく、主にしきい値によって変化することがわかった。

4.3 2段階の統語処理法との比較

機能語予測 CYK 法と、文節検出+統語処理の2段階による認識との比較を行った。2段階の処理において、文節の検出には構文駆動型連続 DP 法⁽¹⁵⁾を用いた。構文駆動型連続 DP 法ではパラメータとしてしきい値が与えられる。また、統語処理には CYK 法に基づくアルゴリズム⁽²²⁾を用いた。このアルゴリズムではパラメータとしてビーム幅が与えられる。一方、機能語予測 CYK 法では、パラメータとしてしきい値とビーム幅の両方が与えられる。そこで、この二つのパラメータを変えて実験を行った。まず、ビーム幅を 5, 10, 15 に固定し、その条件下でしきい値を変えながら認識実験を行い、単語認識率を算出する。この場合の単語認識率は、実質語・機能語のいずれも 1 単語として算出し

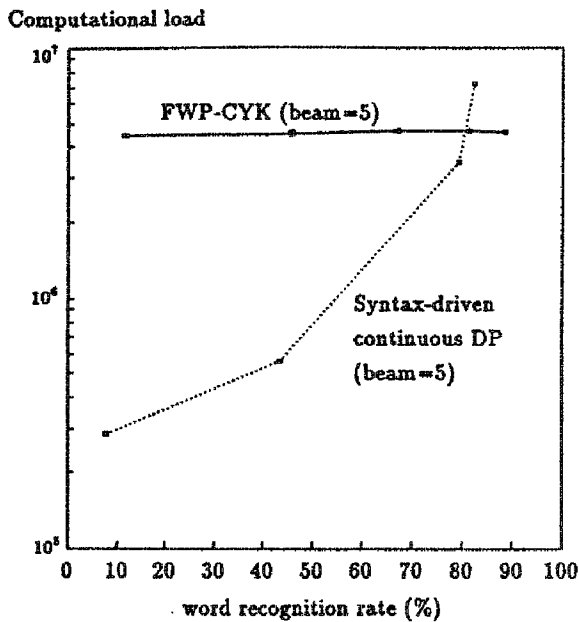


図8 構文駆動型連続DP法と機能語予測CYK法のDPマッチングの計算量比較

Fig. 8 Computational load of the DP matching.

たものである。しきい値を変えることによって単語認識率が変化するから、この単語認識率をパラメータとしてDPマッチングと統語処理の計算量の比較を行った。結果の図中で、単語認識率が最高85~90%程度となっているが、実験に用いたシステムの計算時間・記憶容量の関係上、これ以下までしきい値を下げた実験は困難であった。

はじめに、機能語予測CYK法と構文駆動型連続DP法について、計算量の比較を行った。ここでは、計算量の目安として、DPマッチングの漸化式計算の回数を用いる。それぞれの方法について、単語認識率と1文当りの漸化式計算回数を図8に示す。機能語予測CYK法(図中のFWP-CYK)では、ビーム幅を変化させてもDPの計算量はビーム幅にはほとんど依存しないことがわかる。また、しきい値を変えても計算量はほとんど変わらない。一方、構文駆動型連続DP法では、しきい値を下げることによって急激に計算量が上昇する。そのため、ある程度以上の認識率を得ようとすると、機能語予測CYK法の方が有利であると言える。

次に、前の実験での文節検出によって生成された文節ラティスからCYK法を用いて統語処理を行う方法と機能語予測CYK法を比較した。ここでは、計算量の目安として、依存関係(係り受け)のチェック回数を用いている。それぞれの方法について、ビーム幅を変えた場合の単語認識率と1文当りの依存関係チェック回

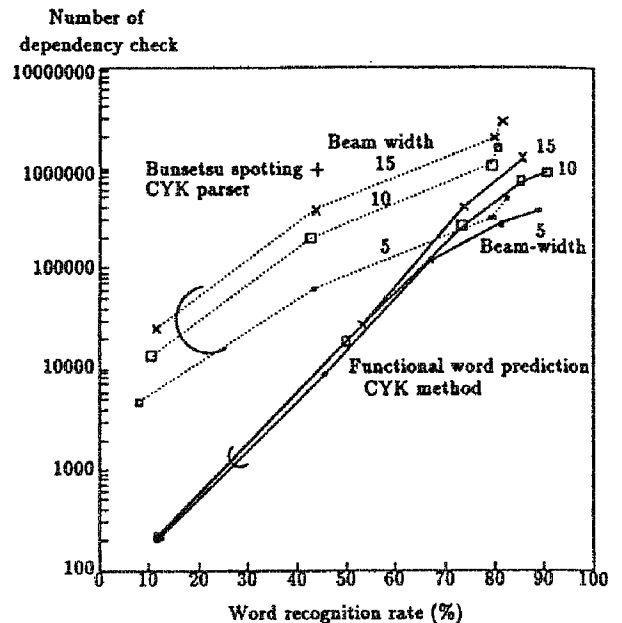


図9 単語認識率に対する統語処理の計算量

Fig. 9 Relation among computational load, parsing method and word recognition accuracy.

数を図9に示す。図中のFWP-CYKは機能語予測CYK法、CYKはCYK法による結果をそれぞれ表している。この結果から、機能語予測CYK法はCYK法に比べて計算量が少ないことがわかる。また、機能語予測CYK法では、ビーム幅の拡大に伴う計算量の増大がCYK法に比べて小さく抑えられている。

これらの結果から、ある程度以上の単語認識率を得る場合(この実験では80%程度)には、2段階の処理方式よりも機能語予測CYK法の方が有利であると言える。

5. む す び

連続音声認識のための新しい統語処理アルゴリズム「機能語予測CYK法」について述べた。この方法は、従来のCYK法に機能語の予測機能を加えたもので、この予測機能によって効率的な処理を行うことができる。また、非終端記号の増加に伴う記憶容量と計算量の増加を抑えるため、機能語予測CYK法にビームサーチを導入したアルゴリズムを提案した。

機能語予測CYK法と、従来の文節検出+統語処理による認識の比較実験を行った結果、機能語予測CYK法はDPマッチングおよび統語処理のいずれの段階においても計算量の点で有利であることが明らかとなった。

今後は、今回比較した従来法以外の方法との比較を行っていくと共に、機能語予測CYK法を実際の連続音

声認識システムに組み込んで、その性能の評価を行っていく予定である。

謝辞 日ごろから御指導頂く、東北大学工学部木村正行教授に深謝致します。

文 献

- (1) Loweire B. T. and Reddy R. : "The Harpy speech understanding system", in "Trends in speech recognition", ed. W. A. Lee, Prentice-Hall (1980).
- (2) Rabiner L. R. and Levinson S. E. : "A Speaker-Independent, Syntax-Directed, Connected Word Recognition System based on Hidden Markov Models and Level Building", IEEE Trans. Acoust Speech & Signal Process., ASSP-33, 3 (1985-06).
- (3) Nay H. : "Dynamic Programming speech recognition using a context-free grammar", Proc. ICASSP87, pp. 69-72 (1987).
- (4) 中川聖一 : "文脈自由文法のフレーム同期型構文解析法による連続音声認識", 信学論(D), J70-D, 5, pp. 908-916 (1987-05).
- (5) 斎藤博昭, 富田 勝 : "LR パーザによる誤りを含む文の認識", 信学技報, SP88-28 (1988-06).
- (6) 北 研二, 川端 豪, 斎藤博昭 : "HMM 音韻認識と予測 LR パーザを用いた文節認識", 信学技報, SP88-88 (1988-10).
- (7) Tomita M. : "An Efficient Word Lattice Parsing Algorithm for Continuous Speech Recognition", Proc. ICASSP86 pp. 1569-1572 (1986).
- (8) 小林 豊, 新美康永, 渦原 茂 : "島駆動音声認識システムにおける言語処理方式", 信学論(D), J71-D, 12, pp. 2563-2570 (1988-12).
- (9) 中川聖一 : "確率モデルによる音声認識", 電子情報通信学会 (1988).
- (10) 岡田美智男, 松尾 広, 伊藤彰則, 小川洋一, 牧野正三, 城戸健一 : "文節スポッティングに基づく日本語文音声の認識", 電学論, 108-C, 10, pp. 826-833 (1988-10).
- (11) 中川聖一, 伊藤立治 : "音節標準パターンと逆時間向き係り受け解析法を用いた日本語文音声の認識", 信学論(D), J70-D, 12, pp. 2469-2478 (1987-12).
- (12) 松永昭一 : "文節音声認識のための係り受けの整合性を考慮した構文解析", 信学論(D), J71-D, 10, pp. 1966-1976 (1988-10).
- (13) 伊藤彰則, 牧野正三, 木村正行, 城戸健一 : "機能語を考慮した連続音声認識のための統語処理アルゴリズムとその評価", 音講論, 3-1-12 (1989-10).
- (14) 首藤公昭, 橋原斗志志, 吉田 将 : "日本語の機械処理のための文節構造モデル", 信学論(D), J62-D, 10, pp. 830-837 (1977-10).
- (15) 岡田美智男, 伊藤彰則, 牧野正三, 城戸健一 : "構文駆動型連続 DP 法による連続音声からの活用語のスポッティング", 信学論(D), J70-D, 12, pp. 2479-2490 (1987-12).
- (16) 岡田美智男 : "アクティブチャート解析法に基づく One-Pass アルゴリズムの構文制御について", 信学技報, SP90-24 (1990-06).
- (17) 嵩 忠雄, 都倉信樹, 谷口健一 : "形式言語理論", 電子情

報通信学会 (1988).

- (18) Gazdar G., Klein E., Pullum G. K. and Sag I. A. : "Generalized Phrase Structure Grammar", Basil Blackwell, Oxford (1985).
- (19) Gunji T. : "Japanese Phrase Structure Grammar", Reidel (1987).
- (20) 伊藤彰則, 小川洋一, 牧野正三, 城戸健一 : "日本語 Dictation System における文節構造モデル", 信学技報, SP87-104 (1987-12).
- (21) 伊藤彰則, 牧野正三, 城戸健一 : "文節認識率に対する音素認識率の変動の影響に関する検討", 音講論, 3-2-7 (1988-03).
- (22) 伊藤彰則, 牧野正三, 城戸健一 : "CYK 法に基づく統語処理アルゴリズム", 音講論, 3-3-12 (1988-10).

(平成 2 年 12 月 19 日受付, 3 年 3 月 19 日再受付)

伊藤 彰則



昭 61 東北大・工・通信卒。昭 63 同大学院修士課程了。現在、同大学院博士課程在学中。音声認識における言語情報処理の研究に従事。日本音響学会、情報処理学会各会員。

牧野 正三



昭 44 東北大・工・電子卒。昭 49 同大学院博士課程了。同年同大電気通信研究所助手。昭 56 同大応用情報学研究センター助手。現在同所助教授。昭 59~61 米国 STL 客員研究員。言語情報を利用した音声認識の研究。音響信号処理、文字認識、画像信号処理の研究に従事。工博。情報処理学会、日本音響学会各会員。

城戸 健一



昭 23 東北大・工・電気卒。昭 38 同大電気通信研究所教授。昭 51 同大応用情報学研究センター教授。平 2, 4 月千葉工業大学情報工学科教授。現在に至る。音響機器、建築音響、騒音制御、心理音響の研究から始まり、現在は音声自動認識、デジタル信号処理、特にその音響工学への応用に関する研究に従事。著書「音響工学」(電子通信学会編, コロナ社), 「デジタル信号処理入門」(丸善), 「電子計算機概論上・下」(丸善), 「過渡現象論」(朝倉書店)等。工博。日本音響学会、電気学会、計測自動制御学会、情報処理学会、韓国音響学会、IEEE, AES など各会員、アメリカ音響学会フェロー。

A Japanese Text Dictation System Based on Phoneme Recognition and a Dependency Grammar

Shozo MAKINO[†], Akinori ITO[†], *Members*, Mitsuru ENDO[†], *Nonmember*
and Ken'iti KIDOT^{††}, *Member*

SUMMARY This paper describes an overview of Japanese text dictation system composed of an acoustic processor and a linguistic processor. The system deals with 843 conceptual words and 431 functional words. The phoneme recognition is carried out using a modified LVQ2 method which we propose. The phoneme recognition score was 86.1 % for 226 sentences uttered by two male speakers. The linguistic processor is composed of a processor for spotting Bunsetsu-units and a syntactic processor. The structure of the Bunsetsu-unit is effectively described by a finite-state automaton. The test-set perplexity of the finite-state automaton is 230. In the processor for spotting Bunsetsu-units, using a syntax-driven continuous-DP matching algorithm, the Bunsetsu-units are spotted from a recognized phoneme sequence and then a Bunsetsu-unit lattice is generated. In the syntactic processor, the Bunsetsu-unit lattice is parsed based on the dependency grammar. The dependency grammar is expressed as the correspondence between a FEATURE marker in a modifier-Bunsetsu and a SLOT-FILLER marker in a head-Bunsetsu. The recognition scores of the Bunsetsu-unit and conceptual words were 73.2 % and 85.7 % for 226 sentences uttered by the two male speakers.

1. Introduction

A speech recognizer which can convert a Japanese text utterance to a Kanji-Kana string can be used as a speech word processor and a pre-processor for an automatic interpreting telephony system.

A number of continuous speech recognition systems⁽¹⁾⁻⁽⁶⁾ have been reported. However, there still remain several problems in developing a continuous speech recognition system for ordinary Japanese text utterances. The traditional continuous speech recognition systems only dealt with particular linguistic information in a specified task domain. Therefore, they are not sufficient for processing ordinary Japanese text utterances with a large vocabulary because the previously-developed linguistic processors were based on a sub-set of Japanese language grammar and dealt with a small vocabulary. Several linguistic

processors^{(7),(8)} for ordinary Japanese written texts have been developed in the field of natural language processing. However, they are not effective for dealing with the Japanese text utterances because current acoustic processors give many phoneme recognition errors and those linguistic processors cannot parse an input sentence with many phoneme recognition errors. In order to deal with the Japanese text utterances, we should develop a linguistic processor which deals with ordinary Japanese text utterances containing many phoneme recognition errors and reduces the number of semantically incorrect sentences accepted by the processor to as few as possible. At the same time, the acoustic processor must achieve high performance for recognition of a large vocabulary task. As necessary techniques for the construction of a Japanese text dictation system, we should develop the following methods:

- (1) Phoneme recognition method with high accuracy,
- (2) Bunsetsu-unit spotting method with high accuracy and with a small amount of computation, where the Bunsetsu-unit is a unit which is uttered with one breath and is composed of a conceptual word followed by several functional words, and
- (3) Efficient parsing method taking into account syntactic and semantic constraints.

In this paper, we will describe an overview of the Japanese text dictation system. The system is composed of an acoustic processor⁽⁹⁾ and a linguistic processor which is divided into a processor for spotting Bunsetsu-units⁽¹⁰⁾⁻⁽¹²⁾ and a syntactic processor⁽¹³⁾⁻⁽¹⁵⁾. The syntactic processor makes use of semantic constraints. The acoustic processor yields a phoneme sequence with phoneme recognition errors from input speech. The linguistic processor generates the best sentence hypothesis from the recognized phoneme sequence so as to satisfy syntactic and semantic constraints.

In order to construct the system, we propose a modified LVQ2 method for the phoneme recognition, a syntax-driven continuous-DP for the Bunsetsu-unit spotting and a CYK-based parsing method using semantic constraints for the syntactic processing.

Manuscript received January 8, 1991.

Manuscript revised April 1, 1991.

[†] The authors are with Research Center for Applied Information Sciences, Tohoku University, Sendai-shi, 980 Japan.

^{††} The author is with the Faculty of Engineering, Chiba Institute of Technology, Narashino-shi, 275 Japan.

Finally we will describe the performance of the system when text speech is uttered Bunsetsu by Bunsetsu.

2. Outline of the Japanese Text Dictation System⁽⁹⁾⁻⁽¹⁵⁾

Figure 1 shows a schematic diagram of the Japanese text dictation system. In this research the speech to be recognized includes spoken sentences whose syntax and semantic structures are syntactically and semantically reasonable. We use sentences from a scientific paper, where the sentences contain 843 conceptual words and 431 functional words. Numbers of nouns, conjugated words, adverbs and adnominals are 496, 247, 78 and 22, respectively. Average numbers of Bunsetsu-units and phonemes in a sentence are 4.1 and 47.

Input speech is analyzed by a 29 channel band-pass filter bank. The input speech is represented by a sequence of logarithmic spectra with 10-ms frame shift. In the acoustic processor a phoneme sequence is recognized from the input speech using the modified LVQ2 method⁽⁹⁾ which we proposed.

The structure of Japanese sentences is effectively described by a two-level grammar which consists of an intra-Bunsetsu grammar and an inter-Bunsetsu grammar. Accordingly, the analysis of the Japanese sentences is divided into two stages. The first stage is the extraction of the Bunsetsu-unit candidates from the recognized phoneme sequence. The second one is the analysis of the dependency structure among the Bunsetsu-unit candidates. The Bunsetsu-unit can be modeled by a finite-state automaton which is conve-

nient to describe the syntactic structure. We constructed a finite-state automaton representing the Bunsetsu-unit structure. The test-set perplexity⁽¹⁹⁾ of the finite-state automaton is 230.

The Bunsetsu-unit candidates are spotted from the phoneme sequence using a syntax-driven continuous-DP matching algorithm⁽¹⁰⁾⁻⁽¹²⁾ based on the finite-state automaton mentioned above. In the syntactic processor⁽¹³⁾⁻⁽¹⁵⁾, functional features are given to the Bunsetsu-unit candidates from lexical items, where the functional features contain syntactic and semantic information. We defined 95 functional features. Two adjacent Bunsetsu-units are merged into one unit if they satisfy syntactic and semantic constraints represented in terms of the functional features, and thus a sentence hypothesis which satisfies syntactic and semantic constraints is finally generated.

The proposed linguistic processor can be extended so as to deal with the ordinary Japanese text utterances even if the number of conceptual words is increased.

3. Phoneme Recognition Using a Modified LVQ2 Method⁽⁹⁾

The Learning Vector Quantization (LVQ, LVQ2) methods were proposed by Kohonen et al.⁽¹⁷⁾. McDermott et al.⁽¹⁸⁾ developed a shift-tolerant phoneme recognition system based on the LVQ2 method. In the LVQ2 algorithm proposed by Kohonen, two reference vectors are modified at the same time if the first nearest class to an input vector is incorrect and the second nearest class to the input vector is correct. We propose a modified training method for the LVQ2 method. In

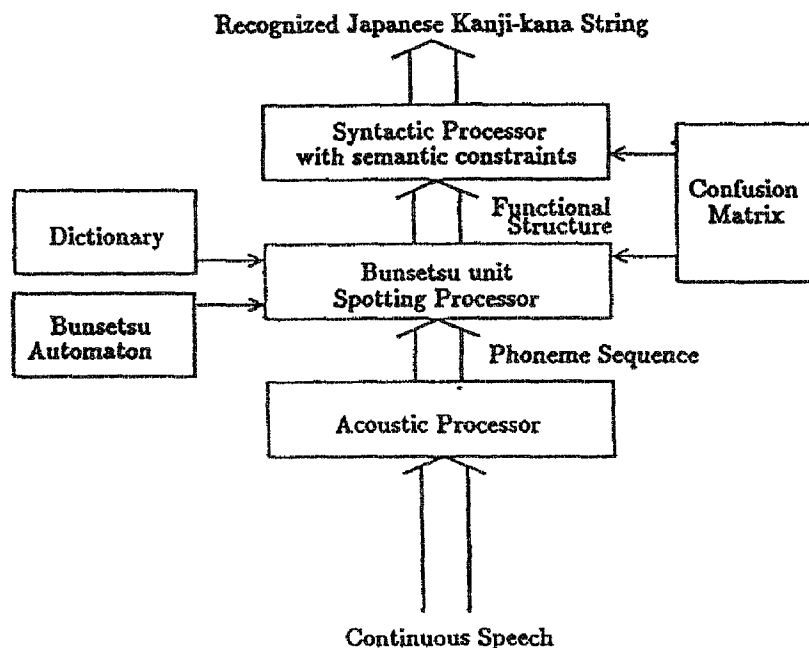


Fig. 1 A schematic diagram of the Japanese text dictation system.

the modified LVQ2 algorithm, n reference vectors are modified at the same time if the correct class is within the N -th rank where N is set to some constant.

Figure 2 shows the process of the modified LVQ2 algorithm. In step 1, reference vectors are chosen using the K -Means clustering method from each class. In step 2, the nearest reference vector of each class to an input vector is selected. In step 3, the rank of the correct class is computed. When the rank of the correct class is n , we assume that the reference vector of the correct class is m_n . In step 4, n is checked to see whether or not n falls in the range of $2 \leq n \leq N$. In step 5, the check is made to see whether or not the input vector falls within a small window, where the window is defined around the midpoint of m_1 and m_n . In step 6, the i -th reference vector is modified according to the following equations.

$$[m_i]^{t+1} = [m_i - \alpha(n)(x - m_i)]^t \quad (i=1, 2, \dots, n-1) \quad (1)$$

$$[m_n]^{t+1} = [m_n + \alpha(n)(x - m_n)]^t \quad (2)$$

The phoneme recognition system is similar to the shift-tolerant model proposed by McDermott et al.⁽¹⁰⁾ as shown in Fig. 3:

- (1) 8 mel-cepstrum coefficients and 8 Δ mel-cepstrum coefficients are computed for every frame from the 29 channel BPF spectrum. The values of each coefficient are normalized by the maximum magnitude of each coefficient. Each reference vector is represented by 112 coefficients (7 frames \times 16 coefficients). Each class was assigned 15 reference vectors chosen by the K -Means clustering method.
- (2) A 7-frame window is moved over the input speech and yields a 112(16 \times 7) dimensional input vector every frame.
- (3) In the training stage the modified LVQ2 method is applied to the input vector as described above.
- (4) In the recognition stage we compute distances between the input vector and the nearest reference vector within each class.
- (5) From this distance measure, each class was assigned an activation value a_w as follows:

$$a_w(c, t) = 1 - d(c, t) / \sum_i d(i, t) \quad (3)$$

where d , c and t are distance, class and time, respectively.

- (6) The final activation a_f is defined by summing 9 activation values as follows:

PHONEME RECOGNITION SYSTEM

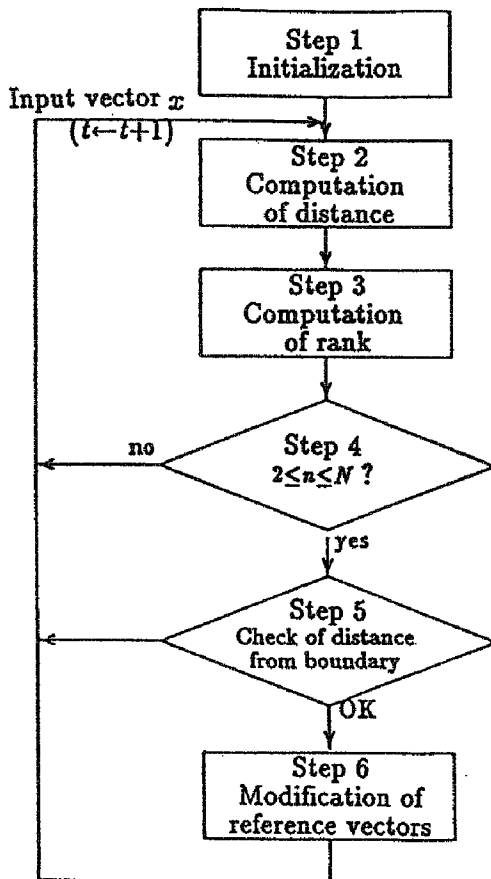


Fig. 2 Algorithm of the modified LVQ2 method.

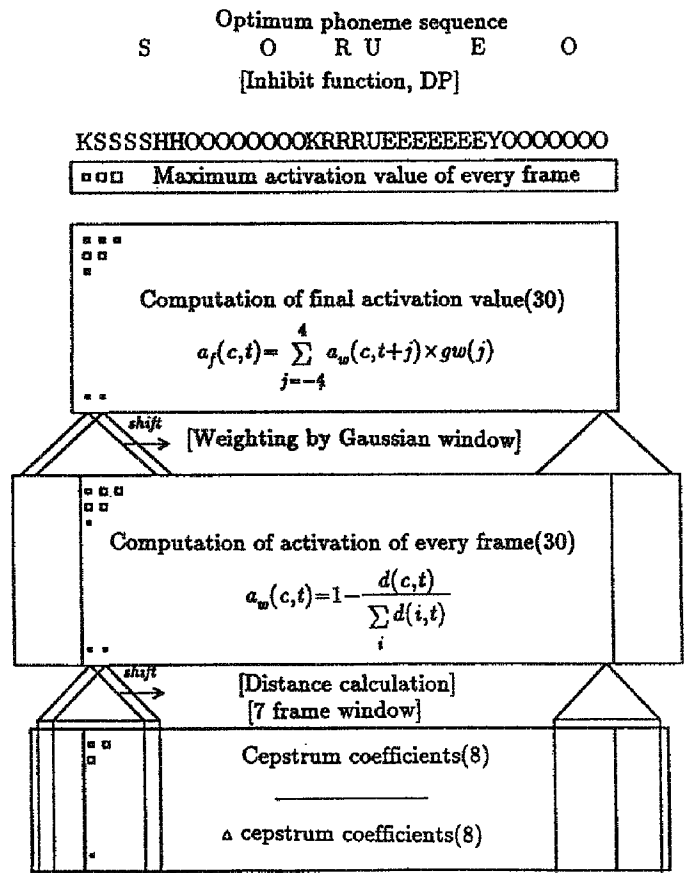


Fig. 3 Phoneme recognition system.

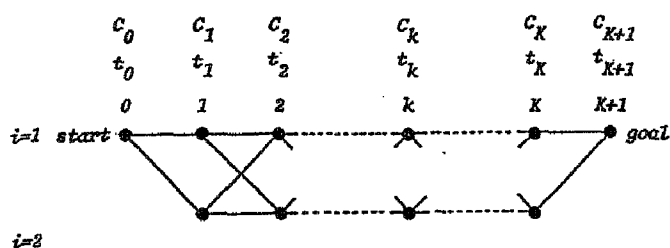


Fig. 4 Network for extracting the optimum phoneme sequence.

$$a_f(c, t) = \sum_{j=-4}^4 a_w(c, t+j) g_w(j) \quad (4)$$

g_w is weight of the gaussian type window.

- (7) A class with the maximum activation value is regarded as phoneme candidate of each frame. The activation value is regarded as an a posteriori probability $P(C_k|t_k)$ of the phoneme C_k at the t_k -th frame.
- (8) The optimum phoneme sequence is computed from the phoneme candidate sequence using the dynamic programming and the duration constraints⁽¹⁶⁾. The phoneme candidate sequence C_k ($k=1, 2, \dots, K$) is assigned to nodes in the first layer ($i=1$) as shown in Fig. 4. Each node at the first layer has a likelihood given by the logarithm of the a posteriori probability $P(C_k|t_k)$ of the phoneme C_k . Each node at the second layer ($i=2$) has the likelihood of error that the phoneme C_k does not occur at the frame position t_k . The likelihood of error is given by the logarithmic value of $\{1 - P(C_k|t_k)\}$. C_0 and C_{K+1} correspond to the initial state and the final state, respectively. The two nodes of C_k and C_r at the first layer should satisfy the duration constraints ($t_k - t_r \geq T(C_k; C_r)$). $T(C_k; C_r)$ is the minimum duration between C_k and C_r . The optimum phoneme sequence is defined as the sequence with the maximum likelihood in all possible paths from the initial state to the final state. Computation of the optimum phoneme sequence is carried out using a dynamic programming method as follows:

$$G_k = \max \left[G_r + \sum_{i=r+1}^{k-1} l_i(2) + l_p(1) \right] \quad (5)$$

$$\begin{aligned} t_k - t_r &\geq T(C_k; C_r) \\ r &> k \end{aligned}$$

where $G_0 = 0$

$$l_k(1) = \log\{P(C_k|t_k)\}$$

$$l_k(2) = \log\{1 - P(C_k|t_k)\}$$

$$l_{K+1}(1) = 0$$

The final result is found in G_{K+1} . By going back through the optimum path, the optimum phoneme

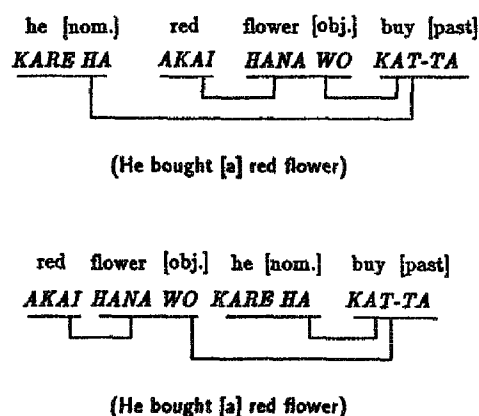


Fig. 5 Examples of the dependency between Bunsetsu-units.

sequence is extracted.

4. Structure of Japanese Sentences

The syntactic features of Japanese language are different from that of English. In Japanese parts of speech can be divided into two classes: conceptual words and functional words. The word order among functional words is fixed and functional words are associated with one conceptual word. Bunsetsu-units can be modifiers, objects, subjects or predicates according to the functional words following the conceptual word. In Japanese, modifier elements precede the modified element. Therefore, subject and object Bunsetsu-units precede predicate, modifier Bunsetsu-units precede the Bunsetsu-unit to be modified, and so on. This dependency between Bunsetsu-units are called "kakari-uke". Some examples of kakari-uke are shown in Fig. 5. Because a kakari-uke structure is another expression of phrase structure tree, lines from a modifier Bunsetsu-unit to a head Bunsetsu-unit should not intersect each other. As long as this principle is followed, the Bunsetsu-unit order is free between modifier Bunsetsu-units which modify one Bunsetsu-unit.

5. Processor⁽¹⁰⁾⁻⁽¹²⁾ for Spotting Bunsetsu-Units

5.1 The Bunsetsu-Unit Structure Model

Because word order between functional words is fixed, the intra-Bunsetsu grammar can be expressed as a finite-state automaton. Figure 6 shows the Bunsetsu-unit structure model for the Japanese text dictation system. The four first arcs, "adverb", "verb/adjective", "noun", "adnominal" represent conceptual words and other arcs represent functional words (or null transitions). Double circles in the figure are the terminal states.

Conceptual words are composed of nouns, verbs/adjectives, adverbs and adnominals. Because

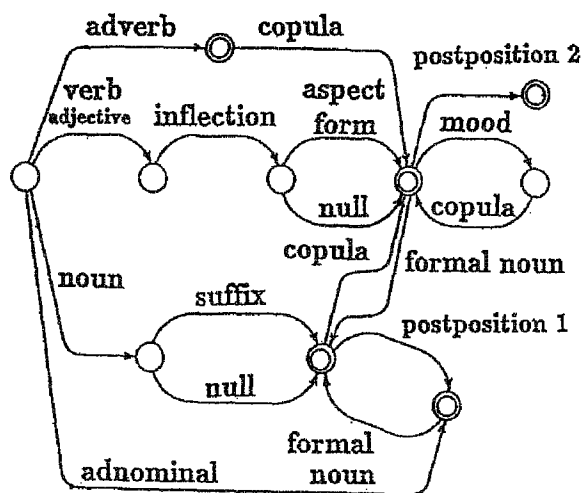


Fig. 6 The outline of the structure of a Bunsetsu-unit.

inflections of verbs and adjectives are similar, they are treated as conjugated words. Auxiliary verbs of voice and negatives can follow a stem of a conjugated word, and then an inflection suffix follows.

Functional words are composed of copulae, aspect forms, modal auxiliaries, formal noun, postposition-1 and postposition-2. Copulae become predicate when they follow nouns or adverbs, and become a part of inflection when they follow stems of modal auxiliaries or "nominal adjectives" (they are also called "adjectival verbs"). Aspect forms express aspect information such as progressive, perfect, etc. Modal auxiliaries express the speaker's point of view. They can be predicates when they follow nouns. Because formal nouns do not appear isolated from conceptual words in a sentence, they can be regarded as functional words.

5.2 Constraints on Bunsetsu-Unit Model

The above-mentioned Bunsetsu-unit structure model is so loosely restricted that it accepts or generates many grammatically incorrect Bunsetsu-units. For example the automaton as shown in Fig. 6 can generate 50×10^9 Bunsetsu-units from a noun. To make the model precise, some constraints are placed on the model.

- (1) Aspect forms are restricted according to aspect features of the preceding verb. There are four aspect classes in Japanese verbs. They are *continuous verbs*, *momentary verbs*, *static verbs* and *fourth verbs*. Static verbs (for example "aru (exist)") express status. Fourth verbs (for example "motozuku (be based on)") also express status and they do not have a concept of time. Aspect classes to be followed can be restricted according to the class of the preceding aspect verb, therefore expressions such as "atte-iru (be existing)" are suppressed.
- (2) Formal nouns and suffixes are restricted accord-

<METHOD 1> Entry all forms of the conjugated word in the dictionary

l_1
KANGAE RARE NA I $l_1 > \theta$

<METHOD 2> Spot a conjugated word by morphemes

l_1 l_2 l_3 l_4
KANGAE RARE NA I $l_1, l_2, l_3, l_4 > \theta$

<METHOD 3> Syntax-driven continuous DP matching algorithm

l_1 l_2 l_3 l_4
KANGAE RARE NA I $l_1, l_2, l_3, l_4 > \theta$

Fig. 7 An example of the Bunsetsu-unit spotting method.

ing to the functional feature of the preceding noun. All nouns have markers which express the functional features used in the syntactic processor.

- (3) Formal nouns are restricted to appear at most once in a Bunsetsu-unit. Formal nouns can appear more than once in a Bunsetsu-unit in real sentences, but such cases are rare.

Using these constraints the number of Bunsetsu-units generated from a noun is reduced to 6.5×10^6 . The number of generated Bunsetsu-units is remarkably reduced compared to the case without the constraints.

5.3 Principle of the Syntax-Driven Continuous-DP Matching Algorithm

There are two traditional methods (methods 1 and 2) for extraction of the Bunsetsu-units as shown in Fig. 7. The first method (method 1) spots the Bunsetsu-units using all possible Bunsetsu-unit reference patterns. However, the method needs a large amount of storage and computation because the number of Bunsetsu-units is huge in the Japanese text dictation system. On the other hand, the second method (method 2) detects the conceptual words and the functional words independently. However, the Japanese language has many functional words with short lengths, such as copulae, inflection of conjugated words and auxiliary verbs. The current spotting method shows poor performance in spotting words having short duration and therefore insertion and deletion errors are common although the amount of computation for this method is very small. The method (method 3) proposed in this paper is an intermediate one. This method spots the Bunsetsu-units based on the finite-state automaton representing the Japanese Bunsetsu-unit structure. We call this method a syntax-driven continuous-DP matching algorithm.

Figure 8 shows an example of processing with the syntax-driven continuous-DP. The processing of the

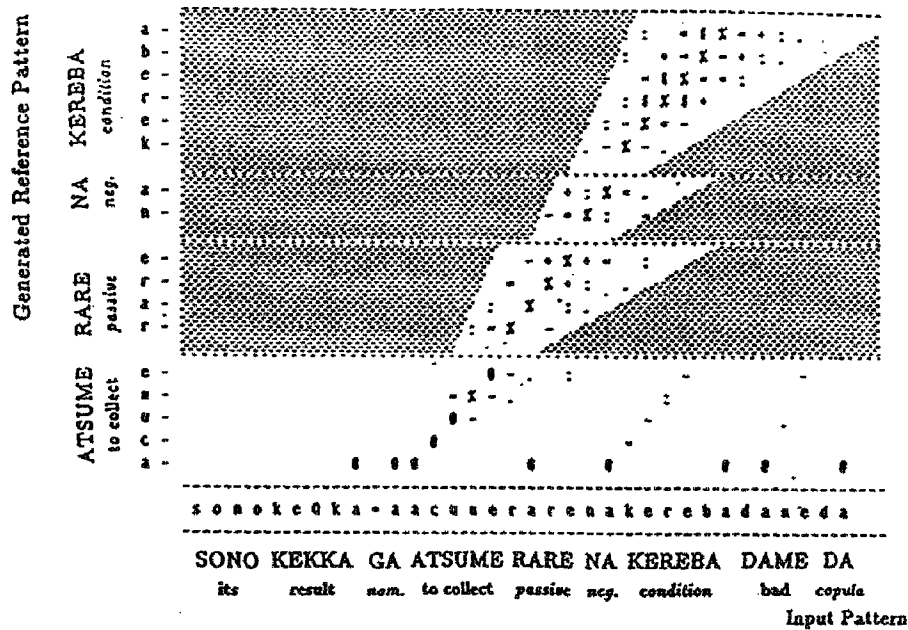


Fig. 8 An example of processing with the syntax-driven continuous-DP.

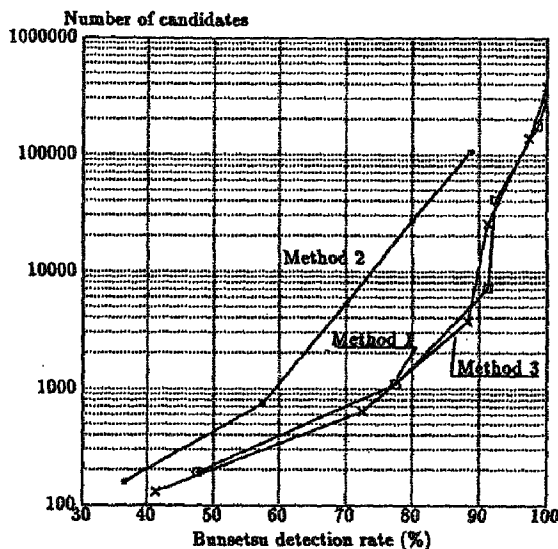


Fig. 9 The relation between the number of candidates per 100 input phonemes and the detection score when the phoneme recognition score is 85%.

syntax-driven continuous-DP starts with a conceptual word. If the likelihood at the final phoneme of the stem of the conceptual word exceeds a threshold, the automaton generates the next word. The calculation of likelihood for the next word is carried out using the final likelihood obtained at the previous stage as the initial value. In the same manner the calculation of the likelihood continues until each path reaches its terminal state as shown in Fig. 6. If the likelihood at the terminal state exceeds a threshold, the Bunsetsu-units of every valid path to that terminal state are recognized as candidates and thus a Bunsetsu-unit lattice is made

from an input phoneme sequence. Using the syntax-driven continuous-DP the Bunsetsu-units are spotted from an input phoneme sequence and simultaneously the morpheme analysis is carried out.

The results in conjugated word spotting can be seen in Fig. 9. This figure shows the relation between the number of candidates per 100 input phonemes and the detection score. Method 3 shows a performance similar to method 1. Method 2 detects 20 times more candidates compared to method 3 when the detection score is 90%.

When a Bunsetsu-unit candidate is detected, a functional structure for the syntactic processor is created. The functional structure contains functional features, most of which are given from lexical items.

6. Syntactic Processor with Semantic Constraints⁽¹³⁾⁻⁽¹⁵⁾

6.1 Parsing by Slot-Filling

The syntactic processing is applied to the candidates of the Bunsetsu-units in the Bunsetsu-unit lattice detected by the syntax-driven continuous-DP. The inter-Bunsetsu grammar is implicitly expressed as the correspondence of the markers in the functional structures of the Bunsetsu-unit candidates. Two partial trees are merged when the modifier's FEATURE marker set and the head's SLOT-FILLER marker set both contain the same syntactic markers. Figure 10 shows an example of the merging of two partial trees. All syntactic dependency, including modification, complement, object and subject are treated in the same framework. We use 95 functional features as the

FEATURE markers and the SLOT-FILLER markers.

6.2 CYK-Based Parsing Algorithm

The algorithm for parsing is based on the Cocke-Younger-Kasami(CYK) algorithm using the beam-search parsing algorithm. Figure 11 shows a parsing table for the CYK-based parsing algorithm. Each box contains Bunsetsu-unit candidates and/or partial trees detected from a phoneme-string interval starting from position i and ending at position j . For example, $X_{5,7}$ contains the Bunsetsu-units and/or the partial trees detected from the interval starting from the fifth

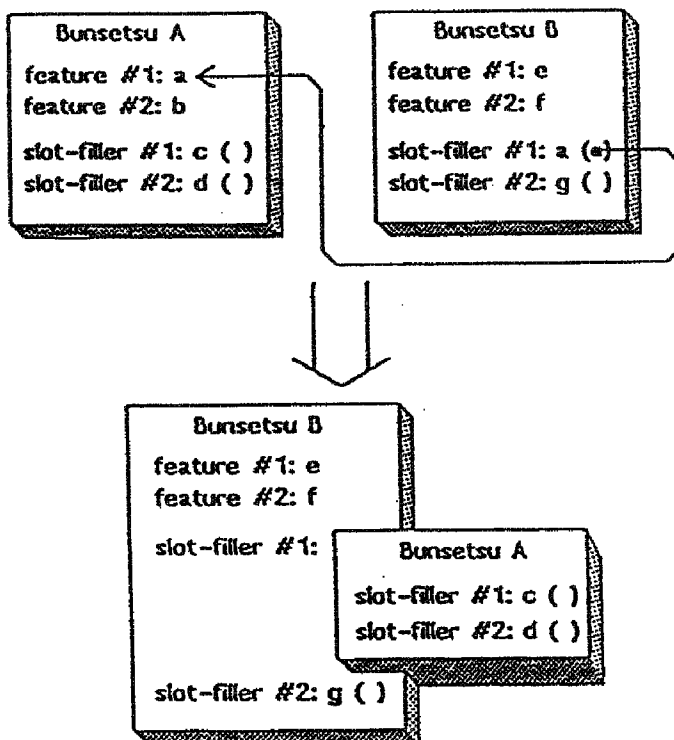


Fig. 10 An example of merging two Bunsetsu-units.

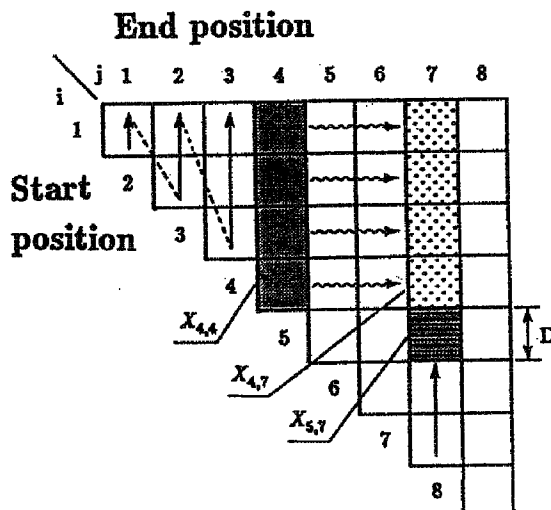


Fig. 11 A parsing table for the CYK-based parsing algorithm.

phoneme and ending at the seventh phoneme. In the parsing, each Bunsetsu-unit candidate is at first stored in the box corresponding to the detected intervals. Assume D be the maximum number of the stored candidates in each box. The candidates are chosen in descending order according to their likelihood values. The parsing proceeds from left to right and from bottom to top in the figure. In this figure, partial trees at $X_{3,7}$ are created from partial trees at $X_{3,4}$ and $X_{5,7}$, partial trees at $X_{4,7}$ are created from $X_{4,4}$ and $X_{5,7}$, and so on. Multiple candidates for a sentence are obtained for one input phoneme sequence using this algorithm. The computation amount for the parsing is $O(N^3 D^2)$, where N is the length of an input phoneme sequence.

7. Experimental Results

The training based on the modified LVQ2 method was carried out for speech samples in the 212 word vocabulary uttered by 7 male and 8 female speakers. The recognition experiments of 30 phonemes were carried out for speech samples in the 212 word vocabulary uttered by another 3 male and 2 female speakers.

Table 1 Speaker-independent phoneme recognition scores for spoken words using the modified LVQ2 method and the method for selecting the optimum phoneme sequence.

Rank of reference vector for training	Phoneme recognition score	Deletion score	Insertion score
N=2	83.1	2.0	11.3
N=3	85.6	1.9	9.8
N=7	86.5	1.7	9.0

Training data: 212 words uttered by 7 male and 8 female speakers

Test data : 212 words uttered by another 3 male and 2 female speakers

Table 2 Multi-speaker-dependent phoneme recognition scores for continuous speech uttered Bunsetsu by Bunsetsu.

Speaker	Phoneme recognition score	Deletion score	Insertion score
A	84.4	4.7	5.8
B	87.8	3.0	9.5

Training data: 70 sentences uttered by 2 male speakers

Test data : the other 226 sentences uttered by the same speakers mentioned above

Table 3 Multi-speaker-dependent Bunsetsu-unit recognition scores for continuous speech uttered Bunsetsu by Bunsetsu.

Speaker	Conceptual word	Bunsetsu-unit	Sentence
A	84.8	70.9	28.4
B	86.7	75.6	36.7

Training data: 70 sentences uttered by 2 male speakers

Test data : the other 226 sentences uttered by the same speakers mentioned above

Table 1 shows phoneme recognition scores. The result of $N=2$ corresponds to the original LVQ2 method. The recognition scores for $N \geq 3$ are higher than the score for $N=2$. This indicates the superiority of the modified LVQ2 method to the original LVQ2 method.

We applied this method to a multi-speaker-dependent phoneme recognition task for continuous speech uttered Bunsetsu by Bunsetsu. Table 2 shows the phoneme recognition scores for 2 male speakers. The training based on the modified LVQ2 method was carried out using 70 sentences uttered by the two male speakers, where each of two speakers uttered 35 sentences. The recognition experiments were carried out for the other 113 sentences uttered by each of the two

Sentence: ニュートンは音響学者ではない
nyutoNwa oNkyogakusyadewanai
Phoneme: nyutoNwa oNkyogakusyadeowawanai
Recognized: ニュートンは音響学者ではない
nyutoNwa oNkyogakusyadewanai

Sentence: 音響現象は流体の振動的力学現象である
oNkyogeNsyowa ryutaino siNdotekirikigakugeNsyodearu
Phoneme: oNkyoNdeNisyuNowa iryukaino sisiNnuokekigirikiryakudeNisyuNdea
Recognized: 音響現象は流体の振動的力学現象だ
oNkyogeNsyowa ryutaino sindotekirikigakugeNsyoda

Sentence: 第2の節目は音響技術に対する電気的应用である
dainino husimewa oNkyogizyucunitaisuru deNkino oyodearu
Phoneme: taiNnido husimewa oNkyoNgizyukinitaisiu teNkino moayodeha
Recognized: 第2の節目は音響技術に対する電気的应用だった
dainino husimewa oNkyogizyucunitaisuru deNkino oyodaQta

Sentence: 特に音の測定には電気はなくてはならない
tokuni otono sokuteniwa deNkiwa nakutewanaranai
Phoneme: hokuN otono sokukuennya teNkiwa nakutewadarana
Recognized: 送る音の測定には電気はなくてはならない
okuru otono sokuteniwa deNkiwa nakutewanaranai

Sentence: 音響に電気はつきものである
oNkyoni deNkiwa cukimonodearu
Phoneme: oNkyoni teNzio cukinonodeae
Recognized: 音響に電気もつきものである
oNkyoni deNkimo cukimonodearu

Sentence: コンピュータもデジタル技術も新しすぎる
konpyutamo dizitarugizyucumo atarasisugiru
Phoneme: koNpyukamo piziuterunizyucumo tatarasuri
Recognized: コンピュータもデジタル技術も新しい
konpyutamo dizitarugizyucumo atarasii

Fig. 12 Examples of sentence recognition.

speakers. The average phoneme recognition score was 86.1 %. The average insertion and deletion scores were 7.7 % and 3.9 %. Table 3 shows recognition scores of the conceptual word, the Bunsetsu-unit and the sentence. The average recognition scores of the conceptual word, the Bunsetsu-unit and the sentence were 85.7 %, 73.2 % and 32.6 %, respectively. Figure 12 shows examples of sentence recognition. Most sentence recognition errors were due to errors in recognition of functional words and in recognition of phonemes at the end of the sentence.

8. Conclusion

We have developed a prototype of a Japanese text dictation system which is composed of an acoustic processor, a processor for spotting Bunsetsu-units and a syntactic processor. We constructed the acoustic processor using the modified LVQ2 method. The modified LVQ2 method achieves a high phoneme recognition performance of 86.1 %. The syntax-driven continuous-DP matching algorithm is used for spotting Bunsetsu-units. This method greatly reduces the computation amount and storage capacity necessary for spotting the Bunsetsu-units. Analysis of dependency structure among the Bunsetsu-unit candidates is effectively carried out using the syntactic and semantic information.

Acknowledgement

The authors wish to thank Dr. Michio Okada at NTT for his helpful suggestions and support in developing the system, and also Prof. Masayuki Kimura for his encouragement.

A part of this work was supported by Grant-in-Aid for Scientific Research on Priority Areas, "Advanced Man-Machine Interface Through Spoken Language", The Ministry of Education Science and Culture, Japan.

References

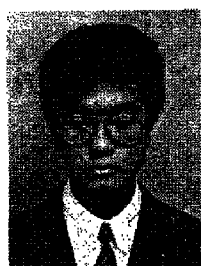
- (1) Sekiguchi Y. and Shigenaga M.: "Speech Recognition System for Japanese Sentences", *J. Acoust. Soc. Japan*, 34, 3, pp. 204-213 (Jan. 1978).
- (2) Shikano K. and Kohda M.: "A Linguistic Processor in a Conversational Speech Recognition System", *Trans. IECE Japan*, J61-D, 4, pp. 253-260 (April 1978).
- (3) Nakagawa S., Ohguro Y. and Hashimoto Y.: "Syntax Oriented Spoken Japanese Recognition/Understanding System —SPOJUS-SYNO—", *Trans. IEICE*, J72-D-II, 8, pp. 1276-1283 (Aug. 1989).
- (4) Tsuboi T., Sugamura N., Tomihisa A. and Obashi F.: "Japanese Conversation Method for Voice Activated Japanese Text Input System", *ibid.*, pp. 1284-1290.
- (5) Matsunaga S.: "Candidate Prediction Using Dependency Relationships Rules Combined with Transition Rules for Minimal Phrase Speech Recognition", *ibid.*, pp. 1299-1306.
- (6) Shigenaga M., Sekiguchi Y., Hanagata T., Taki M. and Yamaguchi T.: "On Prediction Possibility of Predicates and Noun Phrases for Continuous Speech Recognition", *ibid.*, pp. 1307-1312.
- (7) Shudo K., Narahara T. and Yoshida S.: "A Structural Model of Bunsetsu for Machine Processing of Japanese", *Trans. IECE Japan*, J62-D, 12, pp. 872-879 (Dec. 1979).
- (8) Yoshida S.: "Syntax Analysis of Japanese Sentence Based on Kakariuke Relation between Two Bunsetsu", *Trans. IECE Japan*, 55-D, 4, pp. 238-244 (April 1972).
- (9) Endo M., Makino S. and Kido K.: "Phoneme Recognition Using a LVQ2 Method", *IEICE Technical Report*, SP89-50 (Sept. 1989).
- (10) Okada M., Ito A., Matsuo H., Makino S. and Kido K.: "Analysis of Japanese Dictation System", *IEICE Technical Report*, SP86-33 (July 1988).
- (11) Okada M., Makino S. and Kido K.: "A Study of Morphemic and Syntactic Processing Subsystem for Japanese Dictation System", *IEICE Technical Report*, SP86-71 (Dec. 1986).
- (12) Ito A., Ogawa Y., Makino S. and Kido K.: "Refinement and Evaluation of Bunsetsu Automaton in Japanese Dictation System", *Proc. ASJ meeting*, pp. 135-136 (Oct. 1987).
- (13) Ogawa Y., Ito A., Okada M., Makino S. and Kido K.: "Refinement of Syntactic Processor in Japanese Dictation System Using Semantic Information", *Proc. ASJ meeting*, pp. 137-138 (Oct. 1987).
- (14) Ito A., Makino S. and Kido K.: "A Parsing Algorithm Based on CYK Algorithm for Continuous Speech Recognition", *Proc. ASJ meeting*, pp. 91-92 (Oct. 1988).
- (15) Ito A., Makino S. and Kido K.: "Syntactic Processing Using the Principle of Least Bunsetsu's Number Method for Continuous Speech Recognition", *ibid.*, pp. 93-94.
- (16) Moriai S., Makino S. and Kido K.: "A Method for Selecting an Optimum Phoneme Sequence Using a Posteriori Probabilities of Phonemes", *Journal of ASA supplement*, 1, PPP5 (Nov. 1988).
- (17) Kohonen T., Barna G. and Chrisley R.: "Statistical Pattern Recognition with Neural Networks: Benchmarking Studies", *IEEE Proc. ICNN*, 1, pp. 61-68 (July 1988).
- (18) McDermott E. and Katagiri S.: "Shift-invariant Phoneme Recognition Using Kohonen Networks", *Proc. ASJ meeting*, pp. 217-218 (Oct. 1988).
- (19) Jelinek F., Mercer R. L., Bahl L. R. and Baker J. K.: "Perplexity—A measure of difficulty of speech recognition tasks", presented at the 94-th Meet. Acoustical Society of America, Miami Beach, FL (Dec. 1977).



Shozo Makino was born in Osaka, Japan, on January 3, 1947. He received the B. E., M. E. and Dr. engineering degrees from Tohoku University, Sendai, Japan in 1969, 1971 and 1974 respectively. In 1974, he worked with the Research Institute of Electrical Communication, Tohoku University as a Research Associate. In 1980, he worked with the Research Center for Applied Information Sciences, Tohoku University as a Research Associate. Since 1987 he has been an Associate Professor of Information Science at the Research Center for Applied Information Sciences, Tohoku University. His present research interest is in spoken language processing, speech database, image processing and digital signal processing.



Akinori Ito was born in 1963. He received the B. E. and M. E. degrees from Tohoku University, in 1986 and 1988 respectively. He is currently working toward the Dr. Eng. He has been engaged in research on spoken language processing. Akinori Ito is a member of the Acoustical Society of Japan and the Information Processing Society of Japan.



Mitsuru Endo was born in 1966. He received the B. E. degrees from Tohoku University in 1989. He is currently working toward the M. Eng. He has been engaged in research on phoneme recognition using LVQ. He is a member of the Acoustical Society of Japan.



Ken'iti Kido was born in Hamhun, Korea, on April 15, 1926. He received the B. E. and Dr. engineering degrees from Tohoku University, Sendai Japan, in 1948 and 1962 respectively. In 1948, he worked with the Research Institute of Electrical Communication, Tohoku University as a Research Associate. In 1957 he was an Associate Professor in the Faculty of Engineering. In 1963 he was a Professor of Acoustics at the Research Institute of Electrical Communication, and from 1976 to 1990 he was a Professor of Information Science at the Research Center for Applied Information Sciences, Tohoku University, and the Director of the Center. Since 1990 he has been a Professor of Department of Information Engineering at the Chiba Institute of Technology, Narashino, Japan. His present research interest is in digital signal processing and its application to acoustics and speech recognition. Dr. Kido is a Fellow of the Acoustical Society of America. From 1983 to 1985 he was President of the Acoustical Society of Japan.

中川聖一 山本幹雄 田口勝豊 小林聡
(豊橋技術科学大学)

1. はじめに

自然言語の機械処理の研究目的には二つの側面がある。一つは自然言語の現象を解析・記述し、人間の言語理解の過程を説明することである。他の一つは機械翻訳、自動抄録、人間と機械との対話などの実用化を図ることである。但し、何事もそうであるが、両者は明確に区別できるものではない。

我々は、自然言語による質問応答システムの研究を長く行なってきた。¹⁾²⁾³⁾ これは、日本語で書かれた問題文をタイプ入力して、知識データベースを基に答え出力するものであった。しかし、実用化を考える場合、自然言語をタイプで入力するのは思考の即時性を考えると不自然である。自然言語の特性を活かすなら音声を紹介する必要がある。

2. 音声認識のための言語処理⁴⁾

2.1 音声入力とテキスト入力の相異点

テキスト入力を対象とする言語処理の研究は、文理解（質問応答）、テキスト要約、対話システム、かな漢字変換など多方面にわたって研究されてきた。これらのテキストは、構文的にも意味的にも正常で誤りのないとの前提がある。その文字列はあいまい性のない唯一の系列である。

一方、音声入力では誤りのない場合でも、音響処理部と音声認識部を経た出力としての音素（文字）系列は誤りを含んだものとなり、これが言語処理の入力となる。音素認識や音節認識を陽に行なわないシステムでも単語同定や、文節・句同定のレベルで多大のあいまいさがある。テキスト入力に対しても、誤りを含んでいるとの仮定での言語処理が一部研究されているが、音声入力のあいまいさとは比べるべきではない。例えば、単語レベルで考えるなら約10単語からなる文に対する候補出力単語数は1000～5000程度に達する。

このような入力のあいまいさの違いにより、文法・意味情報の利用に際してはテキスト入力に対してはできるだけ正確に入力文の構文・意味表現を抽出するかに重点がおかれ、一方、音声入力に対しては、できるだけ構文的・意味的に非文な文を極力排除するかに重点がおかれる。つまり、テキスト入力文を対象とする場合は、文法が非文を生成するとしても、入力文は非文ではないのであまり問題はないことになる。一方、音声入力の場合には、非文を認識結果としてしまう可能性がある。これこそテキスト入力と音声入力との言語処理における大きな相違点である。図1にこれらの関連を示す。

テキスト入力（かな入力）の場合には、同音異義語などに若干のあいまい性があるのに対し、音声入力のあいまい性は大きい。また、我々の以前の2文節間の係り受けを主体とした言語処理では、任意の文節列（但し、末尾の文節は述語節）の約40%は文として受理された。これらの点を考慮すると、音声理解は自然言語理解よりもより対象を絞ったタスクを扱わざるを得ない。また、タスクで生じうる文のみを生成する文をトップダウン的に作成することが望ましく、タスクに限定されない、音声理解用の文法や意味表現を構築することは大変困難であると言わざるを得ない。勿論、これは、音響処理部や音声認識部（音素・音節認識）の性能に依存することはいうまでもない。また、文中の一部のキーワードだけで文が理解できる場合もあり、すべての場合について、上述のことが言えるわけではない。

bigram, trigram 文法

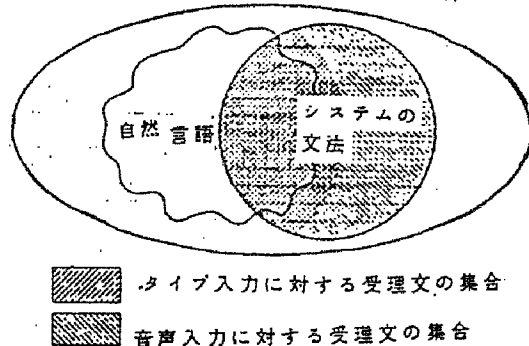


図1 文の集合の関係

2.2 言語の確率モデルの必要性

前節で述べたように、音声認識システムの音韻（単語）認識部の出力結果はあいまいであるために、構文解析の探索空間はテキスト入力時と比べて比較にならないほど膨大になる。そこで、いかに探索空間を小さくするか、逆に言えば自然言語の良い生成モデルをいかに作るかが重要となる。そのためには、大きく分けて次の二つの機能が必要である。

①文法・意味記述の精密化による非文を棄却する能力を高める。

②使用・発声される文の確率・統計的性質を反映させる。

①の機能はテキスト入力文に対しても必要だが、音声入力に対してはさらに留意する必要がある。②の機能も、テキスト入力文に対する構文解析でも有用であることは指摘されている。音声認識でよく使われてきたのは、音素（音節）や単語の二つや三つ組の統計量の使用である。今後は確率文法（単語の共起確率も含めて）の利用が望ましいと考えられる。しかし、いずれにしても、これらの言語の統計的性質を調べるには大規模なデータベースと組織的な研究が必要である。

2.3 音声特有の現象

(a) 韻律情報

韻律情報（ストレス、プロミネンスなど）はスムーズなコミュニケーションには不可欠であり、音声認識システムに組み込むことも必要である。たとえば、ピッチパターンから文節の境界の検出や、文型のタイプ（疑問文／平叙文／命令文）、文構造（係り受け関係）、話題の変化も推定でき探索空間の縮小化に有用である。さらに、話者の意図・感情・発話行為等の理解に利用できよう。

しかし、韻律情報は、音声中の言語情報や話者情報と比べてはるかに不安定で、抽出が困難である。音声を変換する場合、韻律情報はほとんど記述から省略され、わずかに疑問文の「？」や母音を長く発声した場合の「あー」ぐらいである。韻律情報が音声認識にとって有効なのは音声認識システムの音響処理部、音韻（音節）、単語認識部の性能があまりよくない場合とか極めて良い場合であろう。（韻律情報のない平板な音声でも我々は正しく音声を認識・理解できる）。我々の技術レベルはいまちょうどその狭間に達していると思われ、即韻律情報が有効だとは言いきれない。

(b) 不要語および雑音の存在

会話音声には不要語（えー、あー、舌打ち、息つき音など）や背景雑音が存在する。不要語に対しては現象の解析が行なわれているだけで研究例は少ない。最も安易で比較的強力なのはこれらを標

準的パターンとしてシステムに登録することである。また、雑音に関しては様々な研究が行なわれているが、雑音除去処理を行なった後の音声の聴き取りによる理解度は必ずしも向上しないことに、その本質的な難しさがある。雑音に強いパターン照合法や標準パターンの工夫が現実的な解決法であろう。

(c) 非文法的な発話

将来、音声によるマンマシンコミュニケーションの実現を最も困難にする問題の一つは、発声の非文法的現象の取り扱いであろう。例えば、語を言い直したり、言いよどんだりするし、発話途中で、文体が変わり、発話意図の内容が変わることさえある。

このような場合には、意味処理駆動型、島駆動方式、（左から右方向に解析する方法と対照）による解析が有効になろう。小規模なタスクではあるがキーワード検出による音声認識装置が稼働している例がある。しかし、現段階では、このような方式が有効に動作するほどには音声認識の技術は進んでいないと思われる。

3. 対話文の構造分析⁵⁾

3.1 対話資料

対話の分析に使用した対話資料は日本語における各地の観光案内、各種相談などのユーザが情報を得るための対話である。対話方法は2台の計算機を使ったキーボードによるローマ字対話と、音声による対話の2種類である。それぞれの対話資料のデータは以下の通りである。なお、音声による対話資料は日本音聲学会連統音声データベース調査委員会の資料の一部を使用した。

キーボード対話:

ローマ字対話、6名による8対話、1対話当たり20～40ターン（40～80文）、全部で約2,500ターン（約5000文）

発声対話:

37名による36対話、1対話当たり10～80ターン（約30～200文）、全部で約1000ターン（約3000文）

3.2 ネストした対話の分析

対話文の特徴分析は、いろいろ視点からの分析が考えられる。例えば、質問文の形態、倒置表現、間投詞の挿入、助詞の欠落、言いどめ、言い直しなどの表層的な分析も重要である。例えば、助詞の欠落はキーボード対話文1025文中3回であったのに対し、音声対話では3553文中173回生じた。

今回、分析の対象としたネストした部分対話は、質問とそれに対応する応答の対が入れ替わっているものの中で、外側の質問-応答対の両者が入れ替わっているものである。例えば、質問に答える側が質問した場合などである。質問-応答か入れ子になっているパターンを図2に示す。あいづちなどの付加的な発話は無視した。今回対象としたパターンはtype1とtype2である。図中の応答から質問への矢印はその質問-応答の対応を表す。また、[]で囲まれている発話は省略可能であることを示す。

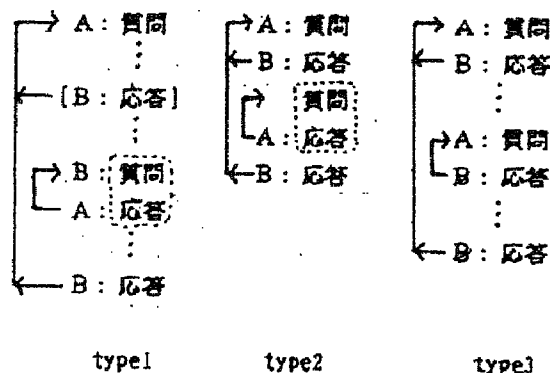


図2 ネストのパターン

また、ユーザがシステムに単なる情報検索でなく、情報の処理を依頼する場合には情報を与える側が主にユーザとなる。例えば、ユーザの希望にあった製品をシステムが見つめるような対話では、システムが主にユーザに対して質問する。このとき、ユーザの最初の依頼を質問と解釈すると、システムの質問がすべてネストしていることになるので、依頼は質問と解釈しなかつた。

図2中の点線で囲まれた部分を一つとして数えた場合、3.1節で述べた対話資料の中にネストした対話は87箇所存在した。ネスト分類を情報を与える側からの質問の目的で分類すると次のようになる。

(1) 詳細化の質問

(1-1) 質問に答えるために必要な情報を得るための質問

(1-2) 質問に対する答えのレベルを決めるための質問

(1-3) 質問の意味または対象が曖昧な場合の質問

(1-4) 応答に対するより詳細な情報を得るための質問

(2) 単語の意味に対する質問

(3) 音声が開き取れなかったために生じる質問

(4) その他

それぞれに分類された対話の数を表1に示す。

表1 ネストの分類

分類	音声		キーボード	
	type1	type2	type1	type2
1-1	50	6	2	0
1-2	1	5	0	0
1-3	9	0	3	0
1-4	6	0	0	0
2	1	0	0	0
3	2	0	-	-
4	2	0	0	0

まず全体のターン数に比べてネストしている対話は非常に少ないことが言える。特にキーボード対話では少ない。これは、ネストの定義で意味的な要素をかなり排除したためであると思われる。意味的なネスト（談話構造）を考えると各質問-応答対は何らかの関係を持っているため、ほとんど全ての対がネスト構造を持っていると言つてよい。しかし、単純な対話制御部を作成するには、ここで扱ったような表層レベルに近いネストを考慮すれば、十分であると思われる。

ネストしている対話の約90%が質問に答えるための詳細化の質問であることが分った。音声が開き取れないための確認の質問は当初、多いであろうと予想していたが、ほとんど生じていない。しかし、システムが音声認識装置に基づくとなれば、無視できない重要な問題となろう。

4. 対話システム⁶⁾

我々は音声による自然言語対話システムを構築中である。アプリケーションとしては、ユーザがある情報を観光案内のデータベースから検索するシステムを考えている。このシステムを単なる質問応答ではなく、より自然な対話を行なえるようにするための一段階として、情報を与える側（多くの場合はシステム）からの質問を許した対話を考えている。

また、対話における文脈レベルでの曖昧さは人間にとっても解決するのが困難な場合があり、その場合には発話の意味に関する確認・明白化の質問、または詳細化の質問とそれに続く対話が必要となる。計算機による自然言語対話システムを考えた場合、現状では曖昧性を解決する能力が人間より悪いため、システムが解決できない曖昧性は積極的な人間に質問するようなシステムを考えなければならぬ。

ここでは、まずキーボード入力による計算機との対話システムについて述べる。

4.1 対話システムの全体構成

図3に対話システムの全体構成図をしめす。

対話システムは、日本語解析部、対話処理部の処理系と日本語辞書、日本語文法辞書の辞書部、そして観光案内のための知識データベースから構成されている。日本語解析、日本語辞書、文法辞書は以前我々が開発した質問応答システム³⁾とほとんど同じものを使用している。質問応答システムでは、入力文は交通規制の平叙文による問題文であり、この問題文に対して答えを推論していた。対話システムでは平叙文以外に疑問文もでてくる（命令文は考えない）。そこで対話システムでは文法規制を拡張して疑問文の解析を行なえるようにしている。さらに、意味解析結果である意味ネットワークに対して疑問文を表現するための概念やリンクを付け加えた。

次に対話システムの処理の流れについて説明する。対話システムにユーザから入力があると、日本語解析部で構文・意味解析を行い意味ネットワークに変換する（このとき意味解析結果は正しいもので唯一であり、必須各の省略がないものとする）⁷⁾。

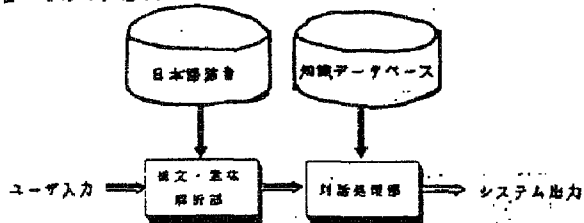


図3 対話システム全体構成図

つぎに入力文の種類の識別を行ない、疑問文であればそれに対する応答文を生成する。生成できないは対話ルールインタプリタを起動して対話文を生成する。

入力文が平叙文であるとき、この入力文がシステムと質問に対する応答であるときには応答から必要な情報を取り出して文脈情報に蓄える。そして、対話ルールインタプリタを起動して対話文を生成する。入力文がシステムに対する応答でないときも対話ルールインタプリタを起動して対話文生成を行なう。

本稿では、本システムの特徴である対話ルールとそのインタプリタについて述べる。

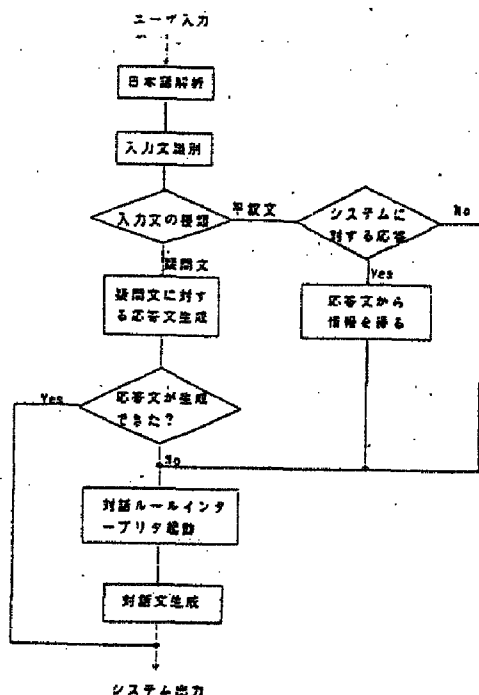


図4 対話システムの処理流れ図

4.2 対話インタプリタ

(a) 対話ルール

対話ルールは、図5の構文にしたがって図6のように記述される。この対話ルールは状況に応じてアドホックに書かれている。

この対話ルールは対話が始ったときに、必ず一つは選ばれていなければならない。対話システムは、ユーザの入力から対話の目的を認識して、目的にあった対話ルールを選択する。そして、対話ルール用スタックに蓄える。例えば、この対話システムは観光案内を対象としているので、“～へ行く”、“～について知る”というような目的がほとんどである。“～へ行く”のような目的のときには図6に示されているような対話ルール“R_行く”が選択される。

対話ルールが選択されたとき、対話ルールのスロットは必ず埋められなければならない。スロットに埋められる値は入力文の解析結果の意味ネットワークからスロットと同じ属性情報を取り出した値である。

図6の対話ルールの例“R_行く”はスロット値Val1からVal2へ行くための対話ルールで、スロット値Val1, Val2には実際の地名が代入される。例えば“豊橋から富士山まで行く”のような文が入力されたときは、対話ルール“R_行く”が選択され、スロット“From”の値Val1には“豊橋”が、“To”の値Val2には“富士山”が代入される。

この対話ルールの第1番目のルール（If asked交通手段）（tell交通手段）は交通手段について聞かれているとき、知識データベースから交通手段の情報を取り出しユーザに対して“豊橋から富士山”への交通手段を教える。第2のルールは、ルートについて聞かれているとき、システムがユーザの交通手段を知っているときルートを教え、知らないときR_ask交通手段の対話ルールを呼び出し、交通手段を尋ねる。

対話ルールインタプリタに入力があると、インタプリタが起動して対話ルールを解釈・実行する。対話ルールの実行により対話文が生成できれば、そのまま文を出力してユーザの入力待ちになる。対話文が生成できなかったときは、対話ルールが失敗したので対話ルールスタックの一番上の対話ルールを取り去る。そして、“解りません”と出力する。

対話ルールの構文

対話ルール: (対話ルール名 (スロット...))

命令

スロット: ((スロット名 スロット値))

対話ルールの命令群

(if (条件文) (実行1) (実行2))
条件文が真の時実行1を、条件文が偽の時実行2を実行する。

(tell A) ユーザにAに関する情報を教えるための関数

(ask wh A) 疑問詞whを用いてAに関する疑問文を生成する関数

(get a b) データベースから知識を取り出すための関数

(call 対話ルール名) 新しい対話ルールを呼び出す命令

(and 命令1) 命令(条件)を実行して全てが真の時真となる関数

(or 命令1) 命令(条件)を実行して少なくとも一つが真になるとき真となる関数

対話ルールの条件文

(asked A) Aについて質問されている時真となる関数

(know A) 対話によってシステムがユーザから得た知識にAがあるとき真、ないときは偽となる関数

図5 対話ルールの構文と命令

対話ルールの真例

```

(R_行く ((from val1) (to val2)))
  (if (asked 交通手段) (tell 交通手段))
  (if (asked ルート)
      (if (know 交通手段) (tell ルート)
          (call R_ask交通手段 val1 val2)))
  (if (asked 時間)
      (if (know ルート) (tell 時間)
          (tell 時間 all)))
  .
  .
  .
  (if (asked 観光地) (call R_知る val2)))
  .
  .

(R_ask交通手段 ((from val1) (to val2)))
  (and (tell 交通手段)
       (ask which 交通手段))

(R_知る ((loc val1)))
  (if (asked 観光地) (tell 観光地 val1))
  (if (asked 標高) (tell 標高 val1))

```

図6 対話ルールの例

5. むすび

一節で述べたように、音声認識用の言語制約と言語理解用の言語制約は目的が異なる。我々は、現在音声対話システムを構築中であるが、音声認識には、意味を組み入れた文脈自由文法をベースに、言語理解は格解析、係り受け解析をベースに行なっている。

自然言語の意味は、文脈によってはじめて確定する。文脈等の制約を充足する解を見い出す必要がある。一方、音声認識は数百キロビットの情報量をもつ波形から、数百ビットの文字列、数十ビットの意味表現に変換する情報圧縮過程である。これも、語彙、構文、意味、プラズマティクス等の制約を充足する解を見つけることである。一方が、シンボルの世界、他方がパターンの世界であり、言語研究者と音声研究者は水と油といわれるが、両者の融合化が是非必要である。

謝辞

本研究の一部には、文部省科学研究費試験研究(B)「課題番号02555067」、一般研究(B)「課題番号03452187」、東海産業技術振興財団から援助を受けた。

参考文献

- 1) 山本、中川：日本語の算数文章題の理解・解決システム、情報処理学会「自然言語処理技術シンポジウム」論文集pp. 59-68(1984)
- 2) 中川、千田：交通規則文章題の理解と質問応答、電子情報通信学会論文誌、Vol. 70D, No. 11, pp. 2280-2286(1987)
- 3) 中川、竹本、田口：交通規則文に関する質問応答システムLICE NCEにおける日本語文から一階述語論理式への変換、情報処理学会論文誌、Vol. 32, No. 3, pp. 354-363(1991)
- 4) 中川聖一：音声認識における言語処理、学術振興会「音声言語・文字言語」委員会、第9回資料、9-3(1988)
- 5) 山本、小林、中川：対話における質問・応答対の入れ子構造の分析、電子情報通信学会秋季大会(1991)
- 6) 田口勝彦：日本語による質問応答・対話システムに関する研究、豊橋技術科学大学修士論文(1991)
- 7) 山本、小林、中川：対話における曖昧さを解消するための質問生成システム、人工知能学会全国大会、13-8(1991)

多重様相論理 TMS

Multi-modal logic TMS

山本 幹雄* 中川 聖一*
Mikio Yamamoto Seiichi Nakagawa

* 豊橋技術科学大学
Toyohashi University of technology, Toyohashi 441, Japan.

1990年3月9日 受理

Keywords: multi-modal logic, truth maintenance system, non-monotonic reasoning.

Summary

Modal logics are primarily concerned with necessity and possibility, but they can also provide a framework for belief, knowledge, time and so on. Although, in general, only one type of modality has been considered at a time, practical knowledge representation languages of AI systems require to represent and manipulate several modalities at the same time. A unified framework for such systems is called multimodal logics. However, automated reasoning in multimodal logics is made difficult. A language for multimodal logics must provide the facilities to support using heuristics and an efficient reasoning mechanism.

The truth maintenance system (TMS) described here can support the reasoning on multimodal logics. It can record justifications for deduced assertions of multimodal logics, track down the assumptions which underlie contradictions, incrementally modify assertional data-base, and calculate the simple deduction about propositional multimodal logics. Our TMS is the extension of McAllester's TMS to multimodal logics. Both his and our TMSs are based on propositional constraint propagation. The difference of them is the objects propagated in TMS. His TMS propagates only truth-value. Our TMS propagates the pair of a truth-value and a set of possible worlds. So our TMS is based on Kripke possible world semantics.

1. はじめに

Truth Maintenance System (TMS)はMcAllesterの定義によれば、以下のような機能を持っているものである⁽¹⁾。

- (1) 事実の集合から命題論理のある形の演繹を行う。
- (2) 演繹の理由を保存し、演繹の結果を説明できる。
- (3) 事実の追加および削除が行われたとき、データベースの一貫性が保たれる。
- (4) 矛盾が生じたときに矛盾の原因となった事実を発見できる。

これらの機能によってTMSは非単調推論やDependency Directed Backtracking (DDb)の実現をサポートする。

McAllesterは上記(1)の特徴を強調し、命題論理の演繹をTMSに明示的に適用できること、TMSを一般的な演繹システムの中の能動的な演繹データベースとして位置づけることができることを示した。結果として彼のTMSは反駁による古典的命題論理の健全な演繹を可能とし、他のTMS(DoyleのTMS⁽²⁾など)では別のコンポーネントが行う矛盾の発見も部分的に可能となる⁽¹⁾。

一方、様相論理は必然と可能性を扱う論理であるが、可能世界モデルの柔軟性によって時間や人間の知識など多種多様の概念のモデルとしても考えることができる。さらに多重様相論理は様相論理を複数の様相演算子を持つように拡張した論理であり、複数の人間の信念・知識や一般的な時間の概念を形式化するための強力な論理システムを提供する。現在、意味論を中心と

した形式化が盛んに研究されている⁽³⁾⁽⁴⁾。しかし様相論理の推論システムは古典論理と比べて一般に複雑であり、DDBのような効率的な推論方式が望まれる。さらに自然言語処理などに応用する場合、対話参加者の信念に関してはもともと不確実な情報しか得ることができないため非単調な推論が必要となる。これらの機能は既存のTMSによっても実現することができるが、既存のTMSは(古典)命題論理のある形の推論しか行わないため、多重様相論理を扱うシステムには不十分である。

本論理では、McAllesterのTMSを多重様相論理に拡張した多重様相論理TMSについて述べる。このシステムは、簡単な命題多重様相論理の演繹と矛盾の発見、演繹と矛盾の理由の説明、および事実が言明・削除されたときにデータベースの一貫性を維持できる。これらの機能によって多重様相論理を扱うシステムを効率的にサポートすることができる。2章で多重様相論理のモデルの枠組みを簡単に説明し、多重様相論理TMSで中心的な役割を果たす世界指標の概念を定義する。3章で多重様相論理TMSの具体的なデータ構造、制約を表すネットワークの生成アルゴリズム、ラベル伝播アルゴリズム、および多重様相論理TMS上での反駁による演繹法を述べる。4章ではTMS上での反駁による演繹の健全性を証明する。

2. 世界指標

2.1 多重様相論理と可能世界モデル

一般の様相論理は一つの様相を形式化するが、一つの枠組みの中で複数の様相を同時に形式化する必要がある場合も多い。例えば、時間の概念を形式化する場合でもFUTURE演算子一つで形式化するのは不十分で、他にNEXT, UNTILなどの様相演算子を必要とする。このような複数の様相を形式化する論理を多重様相論理という。

[1] syntax

まず命題多重様相論理のsyntaxを定義する。

【定義1】 syntax

基本命題論理式の集合 $\Phi = \{P, Q, R, \dots\}$

論理結合子: \neg (否定), \wedge (論理積), \vee (論理和),
 \rightarrow (含意)

様相の集合 $\Sigma = \{a, b, c, \dots\}$

様相演算子: $[m]$, ここで $m \in \Sigma$

命題多重様相論理式: 以下のように再帰的に定義される。

(1) 基本命題論理式は命題多重様相論理式である。

る。

(2) p と q が命題多重様相論理式であるならば,
 $\neg p, p \wedge q, p \vee q, p \rightarrow q, [m]p$ (ただし, $m \in \Sigma$) は
 命題多重様相論理式である。 ■

以下の議論で多重様相論理という場合はこの命題多重様相論理のことを指すことにする。様相演算子 $[m]$ は通常の様相論理の必然性を表す “ \Box ” に相当する。

[2] モデル

多重様相論理のモデルとして一般にKripkeの可能世界モデルを拡張したものが使われる。

マルチフレーム Fr はタプル $\langle W, w_0, P \rangle$ で定義される。ここで W は世界 w_i の集合, w_0 は現実世界を表現しており, $w_0 \in W$, P は W 上での2項関係 $\rho_m (m \in \Sigma)$ の集合である。 ρ_m は様相 m の到達関係, P は到達関係の集合と呼ばれる。一般の多重でない様相論理は一つの様相しか持たず、対応するKripkeモデルは到達関係をただ一つしか持たないため、単純な関係の性質(例えば、推移性、反射性、対称性など)を用いて到達関係を制限することができた。しかし、多重様相論理においては到達関係が複数存在し、各到達関係が独立でない場合がある。例えば、時間に関する様相演算子FUTUREとNEXTは両方が関係する公理(未来すべての真である事実は次の時刻においても真である)を含むため、モデルの各到達関係が相互に依存しあっている。このような依存性を持った到達関係を定義するために、関係の演算である逆関係 (“ -1 ”), 和 (“ \cup ”), 合成 (“ \circ ”) を使って各到達関係の制約を表現する。

例えば、様相 m が推移的である場合 $\rho_m \circ \rho_m \subseteq \rho_m$, 反射的である場合は $\rho_1 \subseteq \rho_m$ (ただし, $\rho_1 = \{(w, w) | w \in W\}$) で表現される。各様相演算子が独立でない場合は二つ以上の到達関係が一つの制限式の中に現れる。例えば、 a, b を様相とし, $[a]p \rightarrow [b]p$ という公理を持つ多重様相論理は $\rho_b \subseteq \rho_a$ という関係の制限を持つモデルによって意味付けされる。多重様相論理の公理から上記のようなマルチフレームの到達関係の制約を決定する方法は文献(3)に述べられている。各到達関係の制約を表す式を到達関係の集合 P の性質と呼ぶ。

マルチモデル M はタプル $\langle Fr, V \rangle$ である。 V は $W \times \Phi$ から $\{T, F\}$ への関数であり、各世界では基本命題論理式の真偽値を決定する付値関数である。付値関数を使って多重様相論理式の真偽値が定義される。

【定義2】 マルチモデル M の世界 w において多重様相論理式 p が真である ($(M, w) \models p$ で表現される) ということは、以下のように再帰的に定義される。

$(M, w) \models p$ iff $V(w, p) = T$

$$(M, w) \models p \wedge q \text{ iff } (M, w) \models p$$

$$\text{かつ } (M, w) \models q$$

$$(M, w) \models p \vee q \text{ iff } (M, w) \models p$$

$$\text{または } (M, w) \models q$$

$$(M, w) \models \neg p \text{ iff } (M, w) \not\models p$$

$$(M, w) \models [a]p \text{ iff } (w, w') \in \rho_a \text{ を満たすすべての } w' \text{ について } (M, w') \models p$$

【定義3】 多重様相論理式 p が妥当である必要十分条件は、すべてのマルチモデル $M \ll W, w_0, p, V$ に関して $(M, w_0) \models p$ が成り立つことである。 ■

2.2 世界指標

多重様相論理での真偽値はマルチフレーム上のある世界を指定して初めて意味を持つ。本論文で述べる多重様相論理 TMS でも世界の集合と真偽値のペアを事実を表現するノードに割り当てる。この節ではマルチフレームの世界の集合を表す世界指標 (world index) を定義する。

【定義4】 世界指標は世界記号 s_i の列 s_1, \dots, s_n である。ここで $s_i \in S = C \cup R$ で、 C は世界定数の集合、 R は世界変数の集合である。世界定数は c_n^m の形をしており、 m は様相の種類を表しており、 ρ_m の m に対応する。 n は世界定数の名前である。名前は簡単に数字で表す。世界変数は v^m の形をしており、 m は様相の種類を表す。 ■

世界指標はマルチフレームの各到達関係がシリアルな関係 (すなわち、各様相 m_i に関して $\rho_i \subseteq \rho_{m_i} \circ \rho_{m_i}^{-1}$ が成り立つ) である場合のみ意味を持つ (4章参照)。したがって、以下のすべての議論では各到達関係はシリアルであるという仮定をおく。

s_1, \dots, s_n が世界指標だとすると、 s_n を $\text{end}(s_1, \dots, s_n)$ と記述する。ある世界記号 s_i の一つ左の世界記号を親記号と呼ぶ。 s_i の左側の世界指標、すなわち s_1, \dots, s_{i-1} を親指標と呼び、parent-index (s_i) と記述する。frame (s_i) は s_i の様相の種類を表す。“+”は二つの世界指標の連結を表す。 $x = s_1, \dots, s_n, y = t_1, \dots, t_m$ とすれば、 $x + y = s_1, \dots, s_n, t_1, \dots, t_m$ となる。

2.3 世界指標の積

世界指標はマルチフレーム上でのある世界の集合を指示している。この節では二つの世界指標が指示する世界の集合に同じ世界がある場合、その同じ世界の部分集合を表す世界指標の積を定義する。二つの世界指標の積は各世界指標の部分列を (後で議論される制約に従って) 他の部分列に書き換えることによって二つの世界指標が同一の世界指標となるとき存在する。

各世界指標の書換えはマルチフレームの到達関係の集合 P の性質によって決定される書換え規則によって制約されるため、まずその書換え規則の決定法の定義をしてから世界指標の積を定義する。

【定義5】 書換え規則

到達関係の集合 P の性質が以下のような形の式の集合であるとする。

$$\rho_{a_1}^{-1} \circ \dots \circ \rho_{a_i}^{-1} \circ \dots \circ \rho_{b_1} \circ \dots \circ \rho_{b_n}$$

$$\subseteq \rho_{c_1} \circ \dots \circ \rho_{c_l}$$

(ただし、 $\rho_{a_i}^{-1}$ と ρ_{b_i} の系列は ϕ でもよい)

性質の各式に対して以下のような書換え規則が導出される (健全性の証明を4章で与える)。

$$S^{a_i}, S^{a_{i+1}}, \dots, S^{a_i}, v^{c_i}, \dots, v^{c_l} \Rightarrow S^{b_1}, \dots, S^{b_n}$$

ここで $a \Rightarrow b$ は a という部分列を b という列に書き換えてもよいことを表す。 v^{c_i} は様相が c_i の世界変数、 S^{a_i} または S^{b_i} は様相が a_i または b_i の世界記号 (すなわち、世界変数でも世界定数でもよい) を表す。また $v^m \Rightarrow c_n^m$ はすべての m に対して常に成り立つ。 ■

【定義6】 世界指標の積

世界指標 x_1, \dots, x_n が与えられたとき、到達関係の集合 P の性質によって (定義5によって) 決定される書換え規則を各世界指標に0回以上適用して得られる世界指標をそれぞれ x'_1, \dots, x'_n とする。このとき世界指標 x_1, \dots, x_n の積が存在する必要十分条件は、 x_1, \dots, x_n に対して $x'_1 = \dots = x'_n$ となるような書換え規則の適用の列がそれぞれ存在することである。 x'_1 を世界指標の積と呼ぶ。ただし、 $n=1$ の場合は必ず積が存在し、 x_1 を世界指標の積とする。 ■

3. 多重様相論理 TMS

3.1 多重様相論理 TMS のデータ構造

多重様相論理 TMS は多重様相論理式の真偽値の制約を表現するネットワークとその上を伝播するラベルデータによって構成されている。ラベルデータは世界指標と真偽値のペア (以下ラベルと呼ぶ) に付加的な情報を加えたものである。ネットワークは式節、制約節、およびリンクからなっている。式節は様相論理の式と伝播されたラベルデータの集合を内部データとして持つ。式節の論理式は各ラベルの世界指標が指示する世界でのラベルの真偽値を持っていることを表現している。制約節はいくつかの式節に対して、リンクによって接続され式節間のラベルの制約を表現する。ある多重様相論理式が TMS に入れられると、いくつかの式節が生成され、さらにその式節の間に制約節とリンクによる制約が張られる。生成された式節は再帰的

に式節と制約節を生成していく。構成されたネットワークを TMS ネット (TMS-net) と呼ぶ。

式節, 制約節, リンク, ラベルデータの詳しいデータ構造を次に定義する。各データ構造は, スロットとその値のペアの集合より構成される。

【定義 7】 多重様相論理 TMS のデータ構造

(1) 式節のスロット

id: 式節のユニークな名前。

formula: その式節が表現する事実の様相論理式。

labels: ラベルデータの集合。

(2) 制約節のスロット

id: 制約節のユニークな名前。

links: リンクの集合。

(3) リンクのスロット

id: リンクのユニークな名前。

f-node: リンクが接続されている式節。

c-node: リンクが接続されている制約節。

link-label: リンクが接続されている式節の真偽値の制約。T または F の値をとる。

link-kind: 値として $+c^m$, $+v^m$, $-$, または nil をとる。各値の説路は 3.3 節で述べる。

mis-align: link-label と不整合しているラベルデータの集合。不整合の定義は 3.3 節で述べる。

active: このリンクを通して伝播されたラベルデータに関する情報を蓄える。実際にはタプル $\langle id, justification \rangle$ の集合であり, id は伝播されたラベルデータの名前, justification は伝播の理由となったラベルデータの名前の集合である。

(4) ラベルデータのスロット

id: ラベルデータのユニークな名前。

label: 世界指標と真偽値のペア。世界指標が支持する世界での真偽値を表す。世界指標を x , 真偽値を tv とすると, $x:tv$ と記述する。真偽値は真を表すときは T, 偽を表すときは F と記述する。

active-links: このラベルデータが伝播されたリンクの集合。

mis-aligned-links: このラベルデータと不整合となるリンクの集合。 ■

各スロットの値はスロット名を関数名とする 1 引数の関数の値として記述できるものとする。引数は基本データの id スロットの値をとる。例えば, 名前が f_{12} という式節の formula スロットの値は $formula(f_{12})$ と記述される。

TMS ネットの例を Fig. 1 に示す。図中の各 TMS ネットは次節で述べる TMS ネット生成アルゴリズムによって生成されたものである。丸は式節を表し, 中

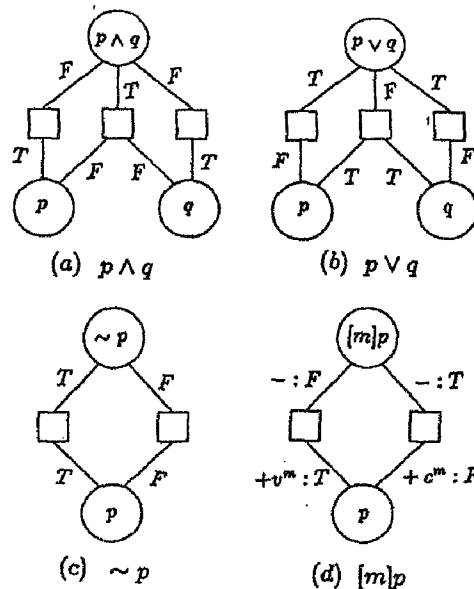


Fig. 1 Examples of a part of TMS-net.

の式は formula スロットの値である。四角は制約節を表す。リンクは式節と制約節を結ぶ線で表され, link-kind と link-label の値を “:” でつないだものを線の横に書いた。ただし, link-kind が nil の場合は link-label のみを書いてある。

3.2 TMS ネットの生成アルゴリズム

いくつかの補助手続きを定義してから前節で述べた TMS ネットの生成アルゴリズムを述べる。

【定義 8】 補助手続き

make-f-node(form): 1 個の式節を生成し, その名前 fn を返す。引数 form は様相論理式である。生成された式節の各スロットの値は次のようになる。

$formula(fn) = form$

$label-data(fn) = nil$

ただし, 以前に同じ引数で式節が生成されている場合は, 新しい式節は生成されず formula スロットの値が同じ式節の id が返される。

make-c-node($link_1, link_2, \dots, link_n$): 1 個の制約節と n 個のリンクを生成し, その制約節の名前 cn を返す。引数 $link_i$ はタプル $\langle f\text{-node}, link\text{-label}, link\text{-kind} \rangle$ であり, 生成されるリンクのスロットの値を指定する。各要素はリンクのスロットの名前に対応し, タプルの値がそのスロットの値となる。例えば, $make\text{-}c\text{-}node(\langle fn_2, T, nil \rangle, \langle fn_4, F, +c^m \rangle)$ は, 一つの制約節とその制約節から fn_2 と fn_4 という名前の式節に接続されるリンクが生成される。各リンクの link-label スロットの値はそれぞれ T と F, link-kind スロットの値はそれぞれ nil と $+c^m$ である。 ■

次に TMS ネットの生成アルゴリズムを示すが、含意記号“ \rightarrow ”は \vee によって間接的に扱うことにし、省略する。直接扱えるように拡張するのは容易である。

TMS ネットの生成アルゴリズム

与えられた多重様相論理式 $form$ から TMS ネットを生成するアルゴリズム $gen-net(form)$ は以下のように再帰的に定義される。 vf_0, vf_1, vf_2 は式節を表す変数とする。

$gen-net(form)$:

(1) $form$ の形によって異なるネットを展開する。

(a) $form$ が $form_1 \wedge form_2$ の形をしている場合

- ① $vf_0 = make-f-node(form)$;
- ② $vf_1 = gen-net(form_1)$;
- ③ $vf_2 = gen-net(form_2)$;
- ④ $make-c-node(\langle vf_0, F, nil \rangle, \langle vf_1, T, nil \rangle)$;
- ⑤ $make-c-node(\langle vf_0, F, nil \rangle, \langle vf_2, T, nil \rangle)$;
- ⑥ $make-c-node(\langle vf_0, T, nil \rangle, \langle vf_1, F, nil \rangle, \langle vf_2, F, nil \rangle)$;

(b) $form$ が $form_1 \vee form_2$ の形をしている場合

- ① $vf_0 = make-f-node(form)$;
- ② $vf_1 = gen-net(form_1)$;
- ③ $vf_2 = gen-net(form_2)$;
- ④ $make-c-node(\langle vf_0, T, nil \rangle, \langle vf_1, F, nil \rangle)$;
- ⑤ $make-c-node(\langle vf_0, T, nil \rangle, \langle vf_2, F, nil \rangle)$;
- ⑥ $make-c-node(\langle vf_0, F, nil \rangle, \langle vf_1, T, nil \rangle, \langle vf_2, T, nil \rangle)$;

(c) $form$ が $\neg form_1$ の形をしている場合

- ① $vf_0 = make-f-node(form)$;
- ② $vf_1 = gen-net(form_1)$;
- ③ $make-c-node(\langle vf_0, T, nil \rangle, \langle vf_1, T, nil \rangle)$;
- ④ $make-c-node(\langle vf_0, F, nil \rangle, \langle vf_1, F, nil \rangle)$;

(d) $form$ が $[m]form_1$ の形をしている場合

- ① $vf_0 = make-f-node(form)$;
- ② $vf_1 = gen-net(form_1)$;
- ③ $make-c-node(\langle vf_0, T, - \rangle, \langle vf_1, F, +c^m \rangle)$;
- ④ $make-c-node(\langle vf_0, F, - \rangle, \langle vf_1, T, +v^m \rangle)$;

(e) $form$ が基本命題論理式の場合

- ① $vf_0 = make-f-node(form)$;

(2) vf_0 の値を $gen-net$ の値として返す。 ■

生成される TMS ネットの一部を Fig. 1 に示す。図中の (a) ~ (d) は生成アルゴリズムの (1) の (a) ~ (d) に対応する。

3.3 ラベル伝播と多重様相論理式の言明

制約節はリンクによって接続されている式節のラベ

ルの制約を表現していると解釈され、制約を満足するようにラベルが伝播される。ラベルの伝播アルゴリズムを説明するためにまず、リンクの整合と多重様相論理 TMS における矛盾の概念を定義する。

【定義 9】 リンクの整合性

あるリンクを l_i 、そのリンクに接続している式節のあるラベルデータのラベルを $x:tv$ とすると、整合性が定義されるのは以下のような場合だけである。

- (a) $link-kind(l_i)$ が nil または “-” である場合
- (b) $link-kind(l_i)$ が $+v^m$ かつ $frame(end(x)) = m$ の場合
- (c) $link-kind(l_i)$ が $+c^m$ かつ $frame(end(x)) = m$ かつ $end(x)$ が世界変数の場合

(a), (b), (c) の場合には以下のように整合性が定義される。

$link-label(l_i) = tv$ であればこのリンクに関して世界指標 x が指示する世界において整合しているという。また $link-label(l_i) \neq tv$ であれば、このリンクに関して世界指標 x が指示する世界において不整合であるという。 ■

【定義 10】 矛盾

ある式節の labels スロットに入っているラベルデータが二つ以上ある場合で、その中のある二つのラベルデータのラベルを $x:tv, x':tv'$ とする。このとき tv と tv' が異なる値で、かつ x と x' の積が存在することを矛盾という。 ■

制約節は接続しているリンクすべてに関して、同じ世界において不整合であってはならないという制約を表現している。すなわち、ある世界においてすべてのリンクに関して整合性が定義されている場合は、少なくとも 1 本のリンクは整合していなければならない。このことから、ラベル伝播の条件が定義される。同時にラベル伝播アルゴリズムの説明で使用されるアクティブリンク、理由ラベル、伝播世界指標も定義する。

【定義 11】 ラベル伝播の条件

ある制約節に接続されているリンクの本数を n とする。

1. $n=1$ の場合

無条件にラベル伝播の条件が満足される。その 1 本のリンクをアクティブリンク、 $\phi: nil$ を理由ラベル、 ϕ を伝播世界指標と呼ぶ。

2. $n \geq 2$ の場合

1 本のリンク l_i を除いた残り $n-1$ 本のリンク $l_i, l_{i-1}, l_{i+1}, \dots, l_n$ に関して、それぞれ世界指標 $x_1, \dots, x_{i-1}, x_{i+1}, \dots, x_n$ において不整合であり、かつ $x_1, \dots, x_{i-1}, x_{i+1}, \dots, x_n$ の積世界指標 z が存在するとき、ラベ

ル伝播が可能であるという。残りの1本のリンクをアクティブリンク, $l_1, \dots, l_{i-1}, l_{i+1}, \dots, l_n$ が不整合となった原因のラベルを理由ラベル, z を伝播世界指標と呼ぶ。

次にあるラベルデータが決定したときのラベル伝播アルゴリズムについて述べる。

ラベル伝播アルゴリズム

新しいラベルデータの追加が決定した式節を fn_0 , 新しいラベルデータの名前を ld とすると, 以下のようラベル伝播アルゴリズム (label-propagation) が再帰的に定義される。

label-propagation (fn_0, ld)

- (1) fn_0 の labels スロットに ld を追加し, labels の他のラベルデータと矛盾していないかチェックする。もし矛盾していれば, 矛盾であることをシステムに伝え label-propagation を再帰のトップレベルまで戻り終了する。
- (2) links を active-link (ld) 以外の fn_0 に接続しているリンクの集合とする。
- (3) links の各要素を調べて, ld の mis-align-links スロットと, 各リンクの mis-aligned-labels を更新する。
- (4) links の各要素であるリンクから接続している各制約節に関してラベル伝播の条件を満たしているかどうか調べる。ラベル伝播の条件を満たしており(そのときのアクティブリンクを $a-link$ とする), かつ link-f-node ($a-link$) が fn_0 と異なる場合には $a-link$ の link-kind スロットの値によって次のようにラベルデータが生成される。伝播世界指標を z とする。
 - (a) link-kind ($a-link$) = nil の場合。
 $\langle id, z: \text{link-label} (a-link), a-link, \phi \rangle$
 - (b) link-kind ($a-link$) = “-” の場合。
 $\langle id, \text{parent-index} (z): \text{link-label} (a-link), a-link, \phi \rangle$
 - (c) link-kind ($a-link$) = $+c^m$ の場合。
 $\langle id, z+c^m: \text{link-label} (a-link), a-link, \phi \rangle$
 ここで, c^m は新しい世界定数である。
 - (d) link-kind ($a-link$) = $+v^m$ の場合。
 $\langle id, z+v^m: \text{link-label} (a-link), a-link, \phi \rangle$
 ここで, v^m は世界変数である。
- (5) (4)で生成された新しいラベルデータをそれぞれ ld_i とすれば, label-propagation (link-f-node ($a-link$), ld_i) をそれぞれラベルデータについて行う。

ある多重様相論理式 $form$ が言明された場合, TMS ネットの生成 gen-net ($form$) を行い, その多重様相論理式を表す式節に1本のリンク (link-label スロットは T) からなる制約節を接続する。1本のリンクからなる制約節は直ちにラベル伝播の条件を満たす, label が $\phi: T$ である新しいラベルデータ ld を作り, そこでラベル伝播アルゴリズム label-propagation (fn, ld) を実行する。矛盾が生じた場合はシステムに矛盾の合図が到達され, アルゴリズムは終了する。

矛盾の原因となった言明の集合を発見する方法と言明を取り消す(削除)アルゴリズムは McAllester の TMS とほぼ同様のアルゴリズムとなるため省略する。リンクの mis-align, active スロットとラベルデータの active-links, mis-aligned-links スロットをラベル伝播が生じたのと逆の方向にたどることによって実行される。詳しくは文献(1)(5)を参照のこと。

3・4 反駁による演繹

(1) 反駁による演繹アルゴリズム

いくつかの多重様相論理式が言明された多重様相論理 TMS 上で反駁による簡単な演繹ができる。反駁による演繹アルゴリズムで証明したい多重様相論理式を p として TMS ネットに展開する。 p を表現する式節に link-label が F である1本のリンクからなる反駁の仮説となる制約節を接続する。 p にラベルが $\phi: F$ なるラベルデータが伝播され, ラベル伝播アルゴリズムによって TMS ネット上をラベルが伝播される。この伝播で矛盾が生じたら反駁によって p が証明されたことになる。証明されたならば, 反駁の仮説となった制約節を取り消して(削除アルゴリズムによって)矛盾を生じさせる原因となる言明から p を真にするような制約節を加える。正確には, 矛盾の原因となる言明の式を表す式節を fn_1, \dots, fn_i, p を表す式節を fn_0 とすると, 次のような制約節を生成する。

make-c-node ($\langle fn_1, F, nil \rangle, \dots, \langle fn_i, F, nil \rangle, \langle fn_0, T, nil \rangle$)

fn_1, \dots, fn_i はラベル $\phi: T$ を持っている(言明されている)ため, fn_0 に接続するリンク以外のすべてのリンクが不整合となり, fn_0 に $\phi: T$ が伝播される。

(2) wise-man パズルの例

複数の人間の知識に関するシステムの例題としてよく用いられる wise-man パズル⁽⁶⁾を例として取り上げる。問題は次のように定義される。「額に白または赤の点を描くが少なくとも1人の額には白い点を描くと王様が n 人の賢者に告げる。実際には全員に白い点を描く。各賢者は他の賢者の額を見ることができが, 自

分の額は見ることができないと仮定する。 $n-1$ 人の賢者が自分の額の色がわからないと言ったところで最後の賢者が自分の色（白）を当てた。これを証明せよ。」

この問題を解くために共有-知識を含む複数の人間の知識を形式化する多重様相論理の公理系を考える。 i は i 番目の賢者の知識, $[E]$ は全員が知っている知識, $[U]$ は共有知識（全員が知っていることを全員が知っている知識）を表す様相オペレータとする。このような多重様相論理系は様相論理のS5という体系を複数の様相演算子に拡張したものに, inductive axiom と呼ばれる公理を追加して形式化される（詳しい公理体系は文献(5)を参照）。

マルチフレームの到達関係の性質と書換え規則は次のようになる。

- (1) $\rho_i \circ \rho_i \subseteq \rho_i: v^i \Rightarrow S^i, S^i$
- (2) $\rho_i \subseteq \rho_i: v^i \Rightarrow \phi$
- (3) $\rho_i^{-1} \subseteq \rho_i: S^i, v^i \Rightarrow \phi$
- (4) $\rho_i \subseteq \rho_E: v^E \Rightarrow S^i$
- (5) $\rho_E \subseteq \rho_U: v^U \Rightarrow S^E$
- (6) $\rho_U \circ \rho_U \subseteq \rho_U: v^U \Rightarrow S^U, S^U$

賢者の数は簡単のため2人(a と b で表し, b をわからないと言ったほうだとする)を考え, 各賢者の知識を表す様相オペレータは $[a]$ と $[b]$ とする（文脈から明らかなため, 賢者の個体定数と知識の様相を同一の記号で表現した）。 $\text{white}(i)$ は賢者 i が白い点を額に持つことを表す。問題を形式化すると次のようになる（問題を解くのに必要なもののみ列記）。

$$[U](\text{white}(a) \vee \text{white}(b)) \quad (1)$$

$$[U](\neg \text{white}(a) \rightarrow [b] \neg \text{white}(a)) \quad (2)$$

$$[U] \neg [b] \neg \text{white}(b) \quad (3)$$

$$[a] \text{white}(a) \quad \{\text{証明すべき命題}\} \quad (4)$$

(1), (2), (3)の言明と(4)の反駁の仮定を表現する多重様相論理TMSネットワークをFig. 2に示す。（“ $p \rightarrow q$ ”は“ $\neg p \vee q$ ”に変換した）。リンクに付いている番号はリンクの名前, 各式節の横に書いてあるラベルはその式節が持っているラベル, ラベルの前の番号はそのラベルを伝播したアクティブリンクの番号を表している。その他はFig. 1の表記に従う。伝播はリンクの番号順に生じたと考えてよい。例えば, 式(3)を表す式節には言明の制約（1本のリンクしか持たない制約節）

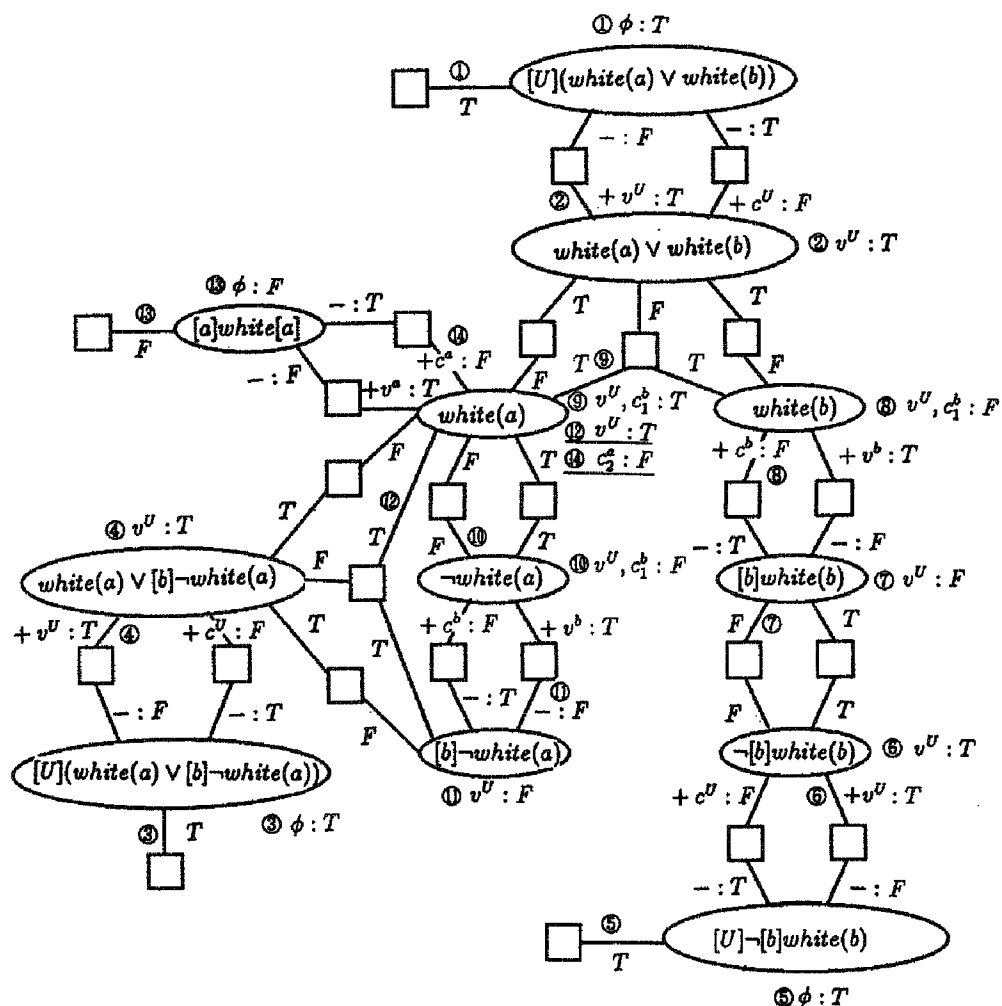


Fig. 2 TMS-net for two-wise-man puzzle.

によって⑤ $\phi: T$ が伝播される。⑤ $\phi: T$ は link-label が F のリンクと不整合となり、次の式節へ⑥ $v^u: T$ が伝播される。このようにして⑧ $v^u, c^b: F$ まで伝播される。⑧ $v^u, c^b: F$ と② $v^u: T$ は⑨のリンクを持つ制約節の2本のリンクを不整合とし、世界指標 v^u, c^b と v^u の積 v^u, c^b が存在するため(書換え規則(4), (5), (6)を使う)ラベル⑨ $v^u, c^b: T$ が white (a) を表す式節へ伝播される。⑨ $v^u, c^b: T$ はリンク⑩を持つ制約節のもう一つのリンクを不整合とし、⑩ $v^u, c^b: F$ が伝播される。このラベルはリンク⑪を持つ制約節のもう一つのリンクと不整合となり、さらにリンク⑪の link-kind が “-” であるため世界指標 parent-index (v^u, c^b) = v^u と F からなるラベル (⑪ $v^u: F$) が伝播される。⑪ $v^u: F$ と④ $v^u: T$ によってリンク⑫がアクティブとなり、⑫ $v^u: T$ が伝播される。最後に white (a) を表す式節上で仮説から伝播してきた⑭ $c^a: F$ と⑫ $v^u: T$ が同時に存在することになる。世界指標 c^a と v^u の積が存在するため矛盾が検出される。よって、命題(4)が真であることが証明される。

4. 多重様相論理 TMS での反駁による 演繹の健全性

3・4節で述べた多重様相論理 TMS での反駁による演繹の健全性を示すために、まず世界指標の積の健全性を示す。世界指標の積の健全性とはすべてのマルチフレームで複数の世界指標が指示する世界が同じ世界を含むときだけ世界指標の積が存在し、積世界指標がその同じ世界の部分集合を必ず指示していることをいう。初めに世界指標が指示する世界を正確に定義する。

【定義 12】 到達世界

マルチフレーム $\langle W, w_0, P \rangle$ 上のある世界の集合 $WS (\subseteq W)$ と、ある W 上の関係 ρ が与えられたとき、 WS から ρ で到達するとができる世界の集合を与える関数 $\Delta_F(WS, \rho)$ は以下のように定義される。

$$\Delta_F(WS, \rho) \equiv \{w | (w', w) \in \rho, w' \in WS\} \quad \blacksquare$$

【定義 13】

世界指標 x を関係に変換する relation (x) は次のように再帰的に定義される。

1. x が空列の場合
relation (x) = ρ_λ
 2. end (x) が世界変数 v^m の場合
relation (x) = relation (parent-index (x)) $\circ \rho_m$
 3. end (x) が世界定数 c^a の場合
relation (x) = relation (parent-index (x)) $\circ \rho_{c^a}$
- ここで、 $\rho_{c^a} = \{(w, f_{c^a}(\{w' | (w, w') \in \rho_m\})) | w \in W\}$,

$f_{c^a}(WS)$ は世界の集合 WS に含まれる一つの世界を返す任意の関数。 \blacksquare

【定義 14】 世界指標が指示する世界

マルチフレーム $\langle W, w_0, P \rangle$ 上での世界指標 x が指示する世界の集合 $\delta_F(x)$ は次のように定義される。

$$\delta_F(x) = \Delta_F(\{w_0\}, \text{relation}(x)) \quad \blacksquare$$

【補題 1】 ある世界指標を x , マルチフレーム Fr の到達関係の集合 P の性質から決定される書換え規則に従って書き換えられた世界指標を x' とすると、 $\delta_F(x') \subseteq \delta_F(x)$ が成り立つ。ただし、各到達関係はシリアルな性質を持っているとする。 \blacksquare

〈補題 1 の証明〉

関係の合成の性質より、

$$\rho_{a_1}^{-1} \circ \cdots \circ \rho_{a_j}^{-1} \circ \rho_{b_1} \circ \cdots \circ \rho_{b_n} \subseteq \rho_{c_1} \circ \cdots \circ \rho_{c_i}$$

ならば

$$\rho_{a_1} \circ \cdots \circ \rho_{a_i} \circ \rho_{a_1}^{-1} \circ \cdots \circ \rho_{a_j}^{-1} \circ \rho_{b_1} \circ \cdots \circ \rho_{b_n}$$

$$\subseteq \rho_{a_1} \circ \cdots \circ \rho_{a_i} \circ \rho_{c_1} \circ \cdots \circ \rho_{c_i}$$

が成り立つ。また、 ρ_{a_i} はシリアルであることから $\rho_\lambda \subseteq \rho_{a_i} \circ \rho_{a_i}^{-1}$ である。さらに $\rho_{a_i} \circ \rho_\lambda \circ \rho_{a_i}^{-1} = \rho_{a_i} \circ \rho_{a_i}^{-1}$ であることから

$$\rho_{a_1} \circ \cdots \circ \rho_{a_i} \circ \rho_{a_1}^{-1} \circ \cdots \circ \rho_{a_j}^{-1} \circ \rho_{b_1} \circ \cdots \circ \rho_{b_n}$$

$$\supseteq \rho_{a_1} \circ \cdots \circ \rho_{a_i} \circ \rho_\lambda \circ \rho_{a_1}^{-1} \circ \cdots \circ \rho_{a_j}^{-1} \circ \rho_{b_1} \circ \cdots \circ \rho_{b_n}$$

$$= \rho_{a_1} \circ \cdots \circ \rho_{a_i} \circ \rho_{a_1}^{-1} \circ \cdots \circ \rho_{a_j}^{-1} \circ \rho_{b_1} \circ \cdots \circ \rho_{b_n}$$

⋮

$$\supseteq \rho_{b_1} \circ \cdots \circ \rho_{b_n}$$

となる。よって、

$$\rho_{b_1} \circ \cdots \circ \rho_{b_n} \subseteq \rho_{a_1} \circ \cdots \circ \rho_{a_i} \circ \rho_{c_1} \circ \cdots \circ \rho_{c_i}$$

さらに、 $\rho_{c^a} \subseteq \rho_m$ であるから上記の式の任意の ρ_{b_i} を ρ_{c^a} に、 ρ_{c^a} を ρ_{a_j} に入れ換えた式も成り立つ。

ここで、ある世界指標を $y, S^{a_1}, \dots, S^{a_i}, v^{c_i}, \dots, v^{c_n} \rightarrow S^{b_1}, \dots, S^{b_n}$ という書換え規則を1段階適用した結果を y' とすると、上記の結果より、relation (y') \subseteq relation (y) となる。よって、 $\delta_F(y') \subseteq \delta_F(y)$ 。

また、すべての体系に成り立つ $v^m \rightarrow c^a$ の規則によって書き換えられた y' に関しても ρ_{c^a} の定義より明らかに $\rho_{c^a} \subseteq \rho_m$ であるから $\delta_F(y') \subseteq \delta_F(y)$ が成り立つ。

書換え規則の適用段階による帰納法により、 $\delta_F(x') \subseteq \delta_F(x)$ が証明される。 \blacksquare

(定理 1) 世界指標の積の健全性

世界指標 x_1, \dots, x_n の積が存在し、その積世界指標を z とすると、すべての到達関係がシリアルであるようなすべてのマルチフレーム Fr で以下の式が成り立つ。

$$\delta_F(x_1) \cap \cdots \cap \delta_F(x_n) \supseteq \delta_F(z) \neq \emptyset \quad \blacksquare$$

〈定理1の証明〉

z はマルチフレーム Fr の到達関係の集合の性質から決定される書換え規則の適用によって各世界指標 x_i から得られるため、補題1より、

$$\delta_F(x_i) \supseteq \delta_F(z)$$

が成り立つ。よって、 $\delta_F(x_1) \cap \dots \cap \delta_F(x_n) \supseteq \delta_F(z)$ 。

マルチフレーム $Fr \langle W, w_0, P \rangle$ の到達関係はすべてシリアルでなければならないという条件より、すべての到達関係 ρ_m の定義域は W に等しい。また ρ_{c_i} の定義より ρ_{c_i} の定義域は ρ_m と等しいことが明らかであるため、すべての ρ_{c_i} の定義域も W である。さらに ρ_λ の定義域は W である。よって、 ρ_m と ρ_{c_i} と ρ_λ の合成よりなる関係の定義域も W である。これは w_0 から到達できる世界が必ず一つはあることを意味しており、すべての世界指標 z に対して $\delta_F(z) \neq \emptyset$ である。

次にある式節とそのラベルから得られた世界指標付き式の定義を行い、世界指標付き式があるマルチモデルで真である必要十分条件を定義する。

【定義15】 世界指標付き式

式節の formula スロットの値を p 、式節の labels スロットの値の中の一つのラベルデータのラベルを x ： tw とすれば、そのラベルの世界指標付き式は x ： p である。ここで、 p は tw の値が T の場合は p 、 F の場合は $\neg p$ である。

【定義16】

マルチモデル $\langle W, w_0, P, V \rangle$ を M 、世界指標付き式を x ： p とすれば、 M で x ： p が真である ($M \models x$ ： p と記述する) とは、 $w \in \delta_F(x)$ を満たすすべての w について $w \models p$ であるとき、またそのときに限る。

この定義から各制約節によるラベル伝播の健全性を示すことができる。

【補題2】 ラベル伝播の健全性

ラベル伝播の条件を満たす理由ラベルの世界指標付き式を x ： p 、 y ： q (制約節のリンクが2本以下ならば、一つだけ考える)、そのラベル伝播の条件でラベル伝播アルゴリズムにより伝播されたラベルデータのラベルの世界指標付き式を z ： p' とする。このとき、すべてのマルチフレーム M に対して、 $M \models x$ ： p かつ $M \models y$ ： q であるならば、 $M \models z$ ： p' である。

〈補題2の証明〉

制約節は \vee と \wedge に関して三つ、 \neg と $[m]$ に関して二つ生成される。さらにアクティブリンクがどれになるかで、3本のリンクを持つ制約節について3種類、2本のリンクを持つ制約節については2種類の伝播に場合分けができる。ただし、 $[m]$ に関する制約節は定義

11より、2本で5種類の伝播が考えられる。結局全部で23通りの伝播の種類に場合分けすることができる。各伝播について健全性を証明すればよい。すべてのラベル伝播について証明するのは紙面の都合上できないので、ここでは代表的な二つの場合に関してのみ証明する。

(1) $p \vee q$ を表す式節が x ： T 、 p を表す式節が y ： F のラベルを持ち、 x と y の積世界指標 z が存在すると、ラベル z ： T が q を表す式節に伝播される。

定理1より $\delta_F(z) \subseteq \delta_F(x) \cap \delta_F(y)$ であるから、すべてのマルチモデル M において理由ラベルの世界指標付き式が真であるとする、 $w \in \delta_F(z)$ なる w においては $(M, w) \models p \vee q$ かつ $(M, w) \models \neg p$ である。よって、多重様相論理式の真偽値の定義より $w \in \delta_F(z)$ なる w においては $(M, w) \models q$ でなければならない。すなわち、すべてのマルチモデル M においてラベル z ： T の世界指標付き式は真である。

同じようにして \vee に関する他のラベル伝播、 \wedge と \neg に関するラベル伝播の健全性が示される。

(2) 式 $[m]p$ を表す式節のあるラベルを x ： T とすると、ラベル $x+v^m$ ： T が式 p を表す式節に伝播される。

すべてのマルチモデル M で、 $M \models x$ ： $[m]p$ であるということは、 $w \in \delta_F(x)$ なるすべての w で $(M, w) \models [m]p$ である。さらに多重様相論理式の真偽値の定義(定義2)より、 $(w, w') \in \rho_a$ なるすべての w' に関して $(M, w') \models p$ である。関数 Δ_F を使うと、 w' の集合は $\Delta_F(\delta_F(x), \rho_m)$ に等しい。ここで δ_F の定義により $\delta_F(x+v^m) = \Delta_F(\delta_F(x), \rho_m)$ であるから、 w' の集合は世界指標 $x+v^m$ によって指示される世界と等しい。よって、すべてのマルチモデル M において、理由ラベルの世界指標付き式が真ならばラベル伝播されたラベルの世界指標付き式も真である。

同じようにして $[m]$ に関する他の四つのラベル伝播の健全性も示すことができる。

補題2より多重様相論理 TMS 上での反駁による演繹の健全性を示すことができる。

(定理2) 多重様相論理 TMS 上での反駁の健全性

p_1, \dots, p_n が多重様相論理 TMS 上で言明されたとする。このとき反駁によって得られた帰結を q とするならば $\vdash p_1 \wedge \dots \wedge p_n \rightarrow q$ が成り立つ。

〈定理2の証明〉

多重様相論理 TMS で p_i が言明された場合、1本のリンクからなる制約節が生成され強制的に世界指標付き式 ϕ ： p_i が TMS 内に導入される。さらに反駁の仮定として世界指標付き式 ϕ ： $\neg q$ が導入される。これ

は、反駁の仮定式 $p_1 \wedge \dots \wedge p_n \wedge \neg q$ を言明したものと全く同じである (\wedge に関する制約節より明らか)。仮定式からの、ラベル伝播によって多重様相論理 TMS 上の矛盾が生じた場合は、定理 1 と矛盾の定義よりすべてのマルチモデルのある世界で矛盾が生じたことを意味する。補題 2 よりラベル伝播による世界指標付き式の妥当性は保持されているため、補題 2 の対偶から仮定式はすべてのマルチモデルで成り立たないことが言える。よって、仮定式の否定 $p_1 \wedge \dots \wedge p_n \rightarrow p$ は妥当である。 ■

5. む す び

本論文では多重様相論理式が扱える TMS を提案

し、その上での反駁による演繹の健全性を示した。

可能世界モデル上のある世界の集合を指示する記号列 (本報告では世界指標と呼んだ) は一般の様相論理で考案されており⁽⁷⁾⁻⁽⁹⁾、本報告の世界指標はそれを多重様相論理に拡張したものである。この (世界の集合を指示する) 記号列を使用する他のシステム (例えば完全な様相論理の証明システム) に本論文で述べた世界指標を使用すれば、容易に多重様相論理へ拡張が可能である。

今後の課題は多重様相論理 TMS を使用した具体的な応用システム (例えば対話システム) の開発、特に非単調論理と DDB を使用したシステムへの応用を考えていきたい。

◇参 考 文 献◇

- (1) McAllester, D. A. : An outlook on truth maintenance, *AI Memo*, No. 551, MIT (1980).
- (2) Doyle, Jon. : A Truth Maintenance System, *Artif. Intell.*, Vol. 12, pp. 231-272 (1979).
- (3) Catnach, L. : Normal multimodal logics, In *Proc. of the 7th AAAI*, pp. 491-495 (1988).
- (4) Halpern, J. and Moses, Y. : A guide to modal logics of knowledge and belief : preliminary draft, In *Proc. of the 9th IJCAI*, pp. 480-490 (1985).
- (5) Charniak, E., McDermott, D. V. and Meehan, J. R. : *Artificial Intelligence Programming* (2nd eds.), Lawrence Erlbaum Associates, Publishers (1987).
- (6) Genesereth, M. R. and Nilsson, N. J. : *Logical Foundations of Artificial Intelligence*, Morgan Kaufmann Publishers, Inc. (1987).
- (7) Jackson, P. and Reichgelt, H. : A general proof method for modal predicate logic, In *Logic-Based Knowledge Representation*, Jackson, P., et al (eds.), MIT Press (1989).
- (8) 米崎 : 様相論理証明器の一般化, 日本ソフトウェア科学会第3回大会論文集, pp. 269-272 (1986).
- (9) Wallen, L. A. : Matrix proof methods for modal logics, In *Proc. of the 10th IJCAL*, pp. 917-923 (1987).

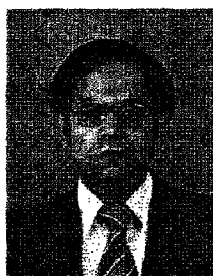
(担当編集委員・査読者：佐藤雅彦)

—著 者 紹 介—



山本 幹雄(正会員)

1984年豊橋技術科学大学情報工学課程卒業。1986年同大学大学院修士課程修了。同年、(株)沖テクノシステムズラボラトリ入社。1988年豊橋技術科学大学情報工学系教務職員。自然言語処理、人工知能に関する研究に従事。情報処理学会、電子情報通信学会、AAAI、ACL 各会員。



中川 聖一(正会員)

1976年京都大学大学院博士課程修了。同年、京都大学工学部情報工学科助手、1980年豊橋技術科学大学情報工学系講師、1983年助教授、1990年教授。1985~86年カーネギーメロン大学客員研究員。工学博士。音声情報処理、自然言語処理の研究に従事。1977年度電子通信学会論文賞受賞。著書「確率モデルによる音声認識」電子情報通信学会(1988)、「音声・聴覚と神経回路網モデル」(共著)オーム社(1990)。

ニューラルネットワークを用いた日本語解析の試み

高 橋 直 人 板 橋 秀 一

ニューラルネットワークを用いた日本語解析の試み†

高橋直人†† 板橋秀一†††

日本語の形態素解析および係り受け解析は、一種の最適化問題、すなわち制約条件を最もよく満たすようにコスト関数の値を最小とする問題、と考えることが可能であり、それゆえに相互結合型ニューラルネットワークで解くことが可能である。この場合、上記のコスト関数はニューラルネットワークのエネルギー関数として表現される。本稿では日本語の形態素解析用と係り受け解析用の2種類のニューラルネットワークの構成、および計算機上でシミュレーションを行った結果について述べる。形態素解析用ネットワークは、分かち書きされていない日本語文を単語単位に切り分ける。複数通りの切り分け方が可能な場合は、文節数最少法と同等の評価関数に基づいて最尤候補を決定し、それを出力とする。係り受け解析ネットワークは文節列を入力とし、それら文節間の係り受け関係を決定する。この際に文法的制約のみでなく、文節間の意味的制約も考慮されるような配慮がなされている。意味的制約は学習用例文中の文節間係り受け頻度に基づいて決定された。どちらの解析においても、パラメータを調整することによって約95%の成功率を得ることができた。入力文の長さが増加しても、ネットワークが収束するまでのステップ数はあまり増加しなかった。

1. はじめに

自然言語文に含まれている曖昧性を解消するためには、統語的解析のみならず、意味的・文脈的・語用論的解析が必要である。人間はこの困難な問題を極めて効率良く行っている。一方、自然言語に関する知識やヒューリスティックスをいかに表現するかという問題にはまだ明確な回答が得られておらず、したがって、計算機で自然言語を解析する場合にはしらみつぶしの方法をとらざるを得ないことが多い。すなわち、文法的に可能なすべての解析木を生成し、それらのなかから与えられた評価関数を最大にするものを選択するのである。評価関数には意味的・文脈的・語用論的妥当性を反映するように構成されたものが選ばれる。このような文法的に可能なすべての解析木の妥当性を評価する方法には、大きな計算コストが必要とされる。

解析対象言語が日本語の場合、形態素解析とは文を単語単位に分割することを指す。日本語文では、通常、単語と単語の間に空白を置かないので、同一の文を単語単位に分割する方法が複数通り存在し得る。特に文が仮名のみあるいはローマ字のみで書かれている場合は分割方法が多数存在するため、組合せの爆発が生じ得る。

一方、日本語の統語的構造は、文節間の係り受け関係で表現することができる。一般的に、文法的制約の

みでは係り受け関係を一意に決定することはできない。ある2文節間の係り受け関係の成立しやすさは、それら2文節に含まれている単語間の意味的關係によって影響を受ける。

形態素解析・係り受け解析の両方とも一種の最適化問題、すなわち与えられた制約条件を最もよく満たすようにコスト関数の値を最小とする問題、と考えることができる。この場合、上記のコスト関数はニューラルネットワークのエネルギー関数として表現される。HopfieldとTankは、ある種の最適化問題は相互結合型のニューラルネットワークを用いることで極めて効率良く解けることを示した^{2),3)}。ニューラルネットワークを自然言語解析に応用した例は幾つか見られるが^{4)-6),8),9),14)}、筆者らの研究¹⁰⁾⁻¹³⁾以外で日本語解析を最適化問題とみなし、それを相互結合型のニューラルネットワークで解いたという例は多くない。自然言語解析をニューラルネットワーク上で実行する方法が確立されれば、近年開発されるようになったニューラルネットワーク・チップと組み合わせることで極めて高速の自然言語解析エンジンの作成が可能となろう。

本稿では日本語文を相互結合型ニューラルネットワークで解析する方法、およびそのニューラルネットワークを計算機上でシミュレートした実験結果について述べる。以下、第2章で相互結合型ニューラルネットワークの一般的な振舞いおよびエネルギー関数について簡単に述べる。第3章では形態素解析に用いたネットワークとその実験結果について、また第4章では係り受け解析に用いたネットワークとその実験結果についてそれぞれ説明する。最後に第5章でまとめと今後の展望について述べる。

† An Experiment of Japanese Sentence Analysis with Neural Networks by NAOTO TAKAHASHI (Doctoral Program in Engineering, University of Tsukuba) and SHUICHI ITAHASHI (Institute of Information Sciences and Electronics, University of Tsukuba).

†† 筑波大学工学研究所

††† 筑波大学電子・情報工学系

2. 相互結合型ニューラルネット

相互結合型ネットワーク内の各ユニット u_i が

$$i_i = \sum_{j=1}^n w_{ij} u_j - \theta_i \quad (1)$$

に対し

$$i_i = \begin{cases} > 0 \text{ then } u_i = 1 \\ = 0 \text{ then 変化せず} \\ < 0 \text{ then } u_i = 0 \end{cases} \quad (2)$$

に従って、非同期に状態遷移を行うものとする。ここで w_{ij} は u_i から u_j への結合の強さ、 θ_i は u_i のしきい値である。この時 w が、 $w_{ij} = w_{ji}$ かつ $w_{ii} = 0$ を満たすならば、このネットワークのエネルギー関数

$$E = -\frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n w_{ij} u_i u_j + \sum_{i=1}^n \theta_i u_i \quad (3)$$

は時間とともに単調減少し、定常状態に収束することが証明できる^{1), 2)}。したがって相互結合型ネットワークを用いてある問題を解くためには、そのネットワークが解に達したときにエネルギーが最低になるように結合の強さおよびしきい値を決定すればよい。

ここで問題となるのは式(3)がエネルギーの最小値ではなくて、極小値に収束する可能性のあることである。この場合は最適解を得ることができない。これを避けるための方法の一つに、ネットワークの動作を確率的にすることがあげられる。Boltzmann Machine は式(2)のネットワークの動作を確率的にしたもので、各ユニットは次のような確率でその状態を1にする。

$$p(u_i = 1) = \frac{1}{1 + \exp(-i_i/T)} \quad (4)$$

式(4)中の T はネットワークの温度と呼ばれる正の数である。 T が0の極限で、Boltzmann Machine は式(2)のネットワークと同じ動作をする。エネルギーの差を強調するためには温度が低い方がよいが、その場合はエネルギーの局所的な谷に陥りやすくなり、最小値付近に到達するまで長い時間がかかる。これを避けるために、初めのうちは温度を高くしておき、状態変化をさせながら徐々に温度を下げる疑似焼きなましの技術を使う。焼きなましのスケジュールは経験的によいものを採用する。

3. 形態素解析

形態素解析ネットワークの各ユニットは、文の構成要素となり得る各単語に対応する。例として「かれがくるまでまつ」という文字列からなる文があり、この

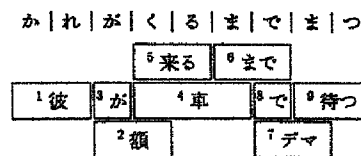


図1 入力文を単語列に分解する方法の例
Fig. 1 An example of dividing a sentence into words.

文に含まれ得る単語としては以下のものがあるとしよう(図1)。

u_1 : 彼(代名詞) u_4 : 車(名詞) u_7 : デマ(名詞)
 u_2 : 額(名詞) u_5 : 来る(動詞) u_8 : で(助詞)
 u_3 : が(助詞) u_6 : まで(助詞) u_9 : 待つ(動詞)

この場合は互いに異なる9単語が存在するので、ネットワーク内のユニット数は $u_1 \sim u_9$ の9となる。ここでの目的はネットワークの活動が収束した状態で値1をとっているユニットに対応する単語のみが文の構成要素となるようにユニット間の重みおよび各ユニットのしきい値を調節することである。簡単のため、以下では「ユニット u_i で表される単語」を単に「単語 u_i 」と表記する。

文を正しい単語列に分割するためには、以下の制約を満たすようなエネルギー関数を作成する必要がある。

1. 入力文中の各文字が、ちょうど1回ずつ用いられるような単語の組合せが選択されたときに最小値をとる
2. 隣接する単語の組合せがすべて文法的に接続可能な場合に最小値をとる
3. 選択された単語列に含まれる自立語の数が少ないほど小さな値をとる

最初の2条件は文法的なものである。最後の条件は、文節数最少法に類似した制限条件を与えるためのものである。上の1~3を表す関数をそれぞれ E_c , E_g , E_l とした場合、形態素解析ネットワークのエネルギー関数はこれらの一次結合

$$E = pE_c + qE_g + rE_l \quad (5)$$

となる。 p , q , r は正の定数である。以下では E_c , E_g , E_l について少し詳しく説明する。

3.1 文字列の分解

n をネットワーク中のユニットの総数、 N を入力文の文字数とする。このとき

$$C_{ik} = \begin{cases} 1 & \text{単語 } u_i \text{ が入力文中で } k \text{ 番目の文字を含む場合} \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

と定義される $n \times N$ 行列 C を考える。図1の例だと

$$C = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix} \quad (7)$$

となる。この C を用いると、入力文中の各文字がちょうど1回ずつ使用されるような単語の組合せが選択された時に最小値0をとる関数は

$$E_C = \sum_{k=1}^N \left(\sum_{i=1}^n C_{ik} u_i - 1 \right)^2 \quad (8)$$

と表すことができる。

3.2 文法的接続可能性

式(8)を用いれば、選択された単語を連結して得られる文字列を入力文と同じものにすることができる。しかし、字面が同一であるからといって、選択された単語列が必ずしも文法的に正しい接続になっているとは限らない。

すべての隣接した単語間の接続が文法的に正しい場合に最小値0をとる式は、以下のように記述することができる。

$$E_G = \sum_{i=1}^n \sum_{j \neq i} G_{ij} u_i u_j \quad (9)$$

ただし

$$G_{ij} = \begin{cases} 1 & \text{単語 } u_i \text{ と単語 } u_j \text{ が入力文中で隣接しており、かつ文法的に接続可能でない場合} \\ 0 & \text{otherwise} \end{cases} \quad (10)$$

である。

例えば「代名詞+名詞」という接続は文法的でないとする。すると図1の u_1 「彼」と u_2 「額」は、文中で隣接しておりかつ文法的に接続可能でないので $G_{12} = G_{21} = 1$ となる。

3.3 自立語数

分ち書きされていない日本語文を単語に分解する際の基準はいろいろ提案されているが、ここでは文節数最少法に類似した自立語数最少法を採用する。すなわち、ある文を分解する方法が複数通りある場合は、そのなかでもっとも自立語数の少ない分解方法を選択するものとする。今回の実験では1文節に含まれる自立語は一つだけとしたので、複合名詞が現れる場合を除けば自立語数最少法と文節数最少法は同じ結果を与える。

まず I_i を次のように定義する。

$$I_i = \begin{cases} 1 & \text{単語 } u_i \text{ が自立語の場合} \\ 0 & \text{その他} \end{cases} \quad (11)$$

この I_i を用いると、式

$$E_I = \sum_{i=1}^n I_i u_i \quad (12)$$

は、選択されたユニット中に自立語が少ないときほど小さな値をとるようになる。

3.4 形態素解析ネットワークのエネルギー関数

以上で E_C , E_G , E_I が決定された。これより式(5)と式(3)を比較すると、形態素解析ネットワークにおけるユニット間の結合の強さ w_{ij} および各ユニットのしきい値 θ_i は以下のようになることがわかる。

$$\begin{cases} w_{ij} = -2 \left(p \sum_{k=1}^N C_{ik} C_{jk} + q G_{ij} \right) \\ \theta_i = -p \sum_{k=1}^N C_{ik} + r I_i \end{cases} \quad (13)$$

3.5 実験

本節では、上で述べたネットワークのシミュレーションプログラムを作成し、実際に文を解析させてみた結果について述べる。計算機は Sun-4/330、言語は Allegro Common Lisp を用いた。入力文は「新明解国語辞典第二版(磁気テープ版)」¹⁵⁾の語義説明文のうち、植物に関する百科事典的記述を抜き出し、平仮名べた書きに改めたものを用いた。平仮名べた書きに改めたのは同音異義語が含まれる可能性を意図的に高くし、それに対してネットワークがどの程度の曖昧性解消能力を持つか見るためである。

まず予備実験として本実験における解析対象文の約1/4にあたる50文を用い、ネットワーク中の各定数をさまざまに変化させて収束の様子を観察した。その結果、式(13)の係数としては

$$p=4, q=2, r=3. \quad (14)$$

という値の組合せが比較的良好な結果を与えることがわかった。

また、時刻 t のときのネットワークの温度 $T(t)$ は次式で与えられるものとした。

$$T(t) = \frac{T_0}{1+t/\tau} \quad (15)$$

ただし

$$T_0=5, \tau=10 \quad (16)$$

である。この T_0 および τ も予備実験によって経験的に得られたものである。

なお、ユニットの初期状態はすべて0とし、状態遷移は非同期に行うこととした。

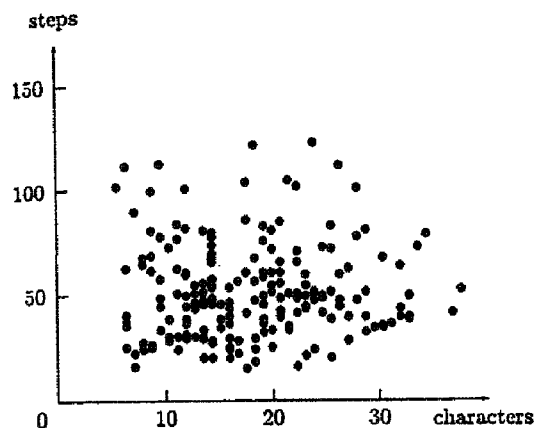


図2 入力文中の文字数と形態素解析ネットワークが収束するまでのステップ数との関係
Fig. 2 Relation between the number of characters contained in input sentences and the steps required to make the network converge.

予備実験で得られた各定数を用いて本実験を行った。本実験の結果を図2に示す。ユニットの状態遷移は非同期かつランダムに生じるので、同一条件で追試を行った場合でも実験結果の細部は異なる可能性がある。ここではすべてのユニットが平均して1回ずつ発火するのに必要な時間を1ステップとしている。なお、約200の入力文に対する解析成功率は約95%であった。

一般に、入力文が長くなるとその中に含まれ得る単語数（すなわちネットワーク内のユニット数）は急激に増加する。しかし図2を見る限り、そのような場合にもネットワークが収束するまでのステップ数はさほど影響を受けないように見える。このように解析時間が入力文の長さあまり影響を受けないという点は、逐次型の自然言語解析システムにはない大きな特徴といえよう。

実験を行った結果、解析途中で一度エネルギーの極小状態に落ちいった場合は、ネットワークが収束するまでのステップ数が増加する傾向にあることがわかった。このことは最終的に正しい解が得られた場合にもそうでない場合にも成り立つ。エネルギーのくぼみ（極小状態）から脱出するためには一度エネルギーが増加する方向へ状態遷移を行う必要がある。くぼみが深い場合、すなわちエネルギー障壁が高い場合はそのような状態遷移を行う確率は小さく、したがってそのくぼみの周辺を長時間「うろつく」ことになるのがこの原因と考えられる。図2中でステップ数が100を越えているものは、ほとんどが一度エネルギー極小状態にとらえられたものである。

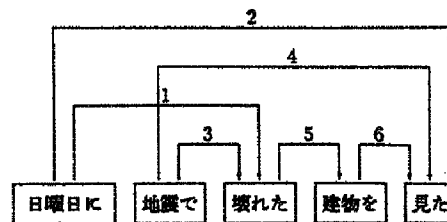


図3 文節間係り受け関係の例(1)
Fig. 3 An example of modification relation of a sentence (1).

4. 係り受け解析

文節間係り受けの条件としては以下の四つが考えられる⁷⁾。

- R1. 最後の文節以外の文節は、自分よりも後方にある文節のどれか一つに係る。
- R2. 係り受け関係は互いに交差しない。
- R3. 文節中の語の意味によって係り受けの成立しやすさが影響を受ける。
- R4. 位置的に近い文節間ほど係り受けが成立しやすい。

R1 より係り受けの総数は構文にかかわらず
(文節数) - 1 (17)

となることがわかる。R2 は非交差条件として有名である。R3 は、例えば図3のような文においては係り受け関係4よりも係り受け関係3の方が成立しやすい、ということを表している。R4 は運用論的な制約であり、統語的あるいは意味的なものではない。

係り受け解析ネットワークがなすべきことは、すべての可能な係り受け関係の中から必要十分な数の関係のみを選択することである。最終的に選択された係り受け関係は上の R1 から R4 を満たしている必要がある。

係り受け解析ネットワークにおいて各ユニットに割り当てられるのは文節そのものではなく、成立する可能性のある係り受け関係である。図3の文の場合には全部で6個のユニットが存在することになる。ネットワークがエネルギー最小状態に収束したとき、値1をとっているユニットが最終的に選択された係り受け関係を表しているものとする。例えば図3の文には解となり得る状態が全部で3通りあるが、そのなかの一つ

unit	u1	u2	u3	u4	u5	u6
value	1	0	1	0	1	1

は、「日曜日に」と「地震で」とが「壊れた」に、「壊れた」が「建物を」に、そして「建物を」が「見た」

に係っている状態を表している*。

一方、状態

unit	u_1	u_2	u_3	u_4	u_5	u_6
value	1	0	0	1	1	1

は係り受け関係1と係り受け関係4とが交差するため、解とはなり得ない。

R1~R4を表す関数をそれぞれ $E_1 \sim E_4$ としたとき、係り受け解析ネットワークのエネルギー関数は

$$E = aE_1 + bE_2 + cE_3 + dE_4 \quad (18)$$

(ただし a, b, c, d は正の定数) で表される。以下に $E_1 \sim E_4$ の定義を与える。

4.1 係り受け関係の数

最終的に値1をとるべきユニットの数は、式(17)より(文節数-1)となる。 n をユニットの総数、 m を最終的に1となるべきユニットの数とすると、望ましい個数のユニットが1となったときに最小値をとる関数は以下のように表すことができる。

$$E_1 = \left(\sum_{i=1}^n u_i - m \right)^2 \quad (19)$$

4.2 相互排他性

各文節の係り先はただ一つである。したがって、係り元が同一の文節であるような係り受け関係が複数個ある場合は、それらのうちの一つだけが値1をとるようにしなければならない。また、非交差条件より、互いに交差するような係り受け関係の両方が1となることは許されない。

このような係り受け関係間の相互排他性を表すために、相互排他行列 X を導入する。 X は $n \times n$ 行列 (n は成立する可能性のある係り受け関係の総数) であり、その (i, j) 要素の値は

$$X_{ij} = \begin{cases} 1 & \text{係り受け関係 } i \text{ と係り受け関係 } j \text{ が} \\ & \text{相互排他的である場合} \\ 0 & \text{otherwise} \end{cases}$$

である。相互排他行列は明らかに対称行列になる。例えば図3の文に対する相互排他行列は

$$X = \begin{bmatrix} 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

* 他の2状態は

unit	u_1	u_2	u_3	u_4	u_5	u_6
value	0	1	1	0	1	1

および

unit	u_1	u_2	u_3	u_4	u_5	u_6
value	0	1	0	1	1	1

である。

となる。

この相互排他行列 X を用いると、値1をとるユニット中のどの二つも相互排他的でない場合に最小値をとる式は

$$E_2 = \sum_{i=1}^n \sum_{j=1}^n X_{ij} u_i u_j \quad (20)$$

と表せる。

4.3 文節間の意味的關係

ある2文節間に係り受け関係が成立するか否かは、係り側の文節と受け側の文節の文法的関係および意味的關係に依存する。係り受け解析ネットワークにおける各ユニットは、もともと文法的に成立し得る係り受け関係のみを表しているの、ここでは文節間の意味的關係を係り受けの成立しやすさに反映させる手段を考えればよい。 i を係り受け関係とし、 Y_i を i という係り受け関係が意味的に成立しやすいほど小さい値をとるものとする。図3の例では「地震で見た」という係り受けよりも「地震で壊れた」という係り受けの方が意味的尤度が高いと考えられるので、 $Y_3 < Y_4$ となるであろう。

この Y_i を用いると、全体として意味的尤度が高い係り受け関係が多く選ばれるほど小さな値をとるような式は

$$E_3 = \sum_{i=1}^n Y_i u_i \quad (21)$$

と表すことができる。

この Y_i をいかに設定するかは重要な問題である。今回の実験では約500の学習用例文から得られた単語間係り受け共起頻度に、式(22)で示す操作を施したものをこの Y_i として用いた。ただし単語間係り受け共起頻度とは、

文節Aが文節Bに係っているとき、文節A中の自立語と文節B中の自立語の単語間係り受け共起頻度を1増やす

という操作を学習用例文全体にわたって施した結果である。

$$Y_i = \begin{cases} 0 & \dots \xi_i = 0 \text{ のとき} \\ -1 & \dots \xi_{\max} = \xi_{\min} \text{ のとき} \\ -\frac{1}{2} \left(\frac{\xi_i - \xi_{\min}}{\xi_{\max} - \xi_{\min}} + 1 \right) & \\ \dots \text{その他.} \end{cases} \quad (22)$$

ただし ξ_i は i 番目のユニットによって示される係り受け関係の単語間係り受け共起頻度であり、

$$\xi_{\max} = \max \{ \xi_i | i=1, \dots, n \}$$

$$\xi_{\min} = \min \{ \xi_i > 0 | i=1, \dots, n \}$$

である。

式(22)は、1) 単語間係り受け共起頻度の符号を反転し、さらに、2) その絶対値が $[0, 1]$ の間におさまるような変形を行う。符号を反転するのは、単語間係り受け共起頻度は意味的尤度が高いほど大きな値をとるのに対し、 Y_i は意味的尤度が高いほど小さい値をとる必要があるためである。

また絶対値を一定の範囲におさめるのは以下の理由による。入力文が異なればその中で用いられる単語も異なる。用いられる単語が異なれば単語間係り受け共起頻度のとる値の範囲も変化する。入力文ごとに意味的尤度を表す値の範囲が変化するのでは安定したパラメータを得ることが困難であるので、どのような入力文に対しても一定の範囲内におさまるようにする必要がある。

式(22)は0以外の値をとる単語間係り受け共起頻度を $[-1.0, -0.5]$ におさまるよう正規化する。ただし、単語間係り受け共起頻度が0のときはそのままに変化させない。これは学習用例文中で一度も係り受け関係を生じなかった単語の組合せをそうでないものから差別化するためである。この操作により、特に入力文中の単語間係り受け共起頻度の範囲が広い場合の解析成功率が向上した。

4.4 文節間の距離

日本語では一般的に、文節間の距離が小さいほど係り受けが生じやすいという傾向がある。係り受け関係 i の係り元を k_i 番目の文節、係り先を l_i 番目の文節とすると、文節間の距離が短い係り受け関係が選ばれらるほど小さな値を取る式は

$$E_i = \sum_{i=1}^n Z_i u_i \quad (23)$$

と表すことができる。ただし、

$$Z_i = l_i - k_i \quad (24)$$

とする。

4.5 係り受け解析ネットワークのエネルギー関数

以上で $E_1 \sim E_4$ が決定された。これより式(18)と式(3)の係数を比較すると、ユニット間の結合の強さ w_{ij} および各ユニットのしきい値 θ_i は

$$\begin{cases} w_{ij} = -2(a + bX_{ij}) \\ \theta_i = (1 - 2m)a + cY_i + dZ_i \end{cases} \quad (25)$$

とすればよいことがわかる。

4.6 実験

形態素解析の場合と同様、本実験の約 1/4 の入力データを使って予備実験を行い、ネットワーク内の定

数を手作業で決定した。

まず、式(18)の a, b, c, d は以下のように決定された。

$$a=6, b=5, c=15, d=1. \quad (26)$$

時刻 t のときのネットワークの温度 $T(t)$ は次の式で与えられるものとした。

$$T(t) = \frac{T_0}{1+t/\tau} \quad (27)$$

ただし

$$T_0=10, \tau=10 \quad (28)$$

である。

また、ユニットの初期状態はすべて0とし、状態遷移は非同期に行った。計算機環境は形態素解析の場合と同一である。

解析対象の文としては、単語間係り受け共起頻度の学習用例文中に現れた単語のみから構成されているものを選んだ。解析対象文・学習用例文はともに文献15)中の植物に関する語義説明文を用いた。形態素解析の場合と異なり、平仮名への変換は行っていない。今回の実験では解析対象文の一部をそのまま学習用例文として用いたが、このように学習用例文中の係り受け関係と解析対象文中の係り受け関係とが類似した傾向のものである場合は、入力した200文のうち95%以上の文において正しい係り受け関係が得られることがわかった。解析した文全体の約5%は、エネルギーの極小値に捕まって脱出できなかった。ユニットの状態遷移は非同期かつランダムに生じるので、同じ文を解析しても異なる結果が導かれる場合もある。

本実験で係り受け解析ネットワークが安定状態に収束するまでに必要とされたステップ数を図4に示す。ここではすべてのユニットが平均して1回ずつ発火するのに要する時間を1ステップとしている。前述のと

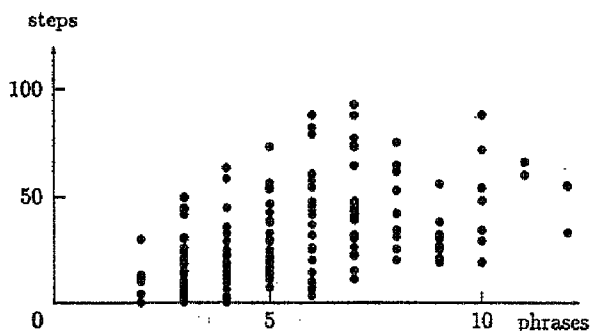


図4 入力文中の文節数と係り受け解析ネットワークが収束するまでのステップ数との関係

Fig. 4 Relation between the number of bunsetsu phrases and the steps required to make the network converge.

おり状態遷移は非同期かつランダムに生じるので、同じ文を解析しても収束までのステップ数が同一になるとは限らないが、全体の傾向はいつまでも図4と大差なかった。図4からは文節数と収束までのステップ数の間に強い相関関係があるとは言えないが、文節数の増加に比してステップ数はそれほど増加しないように見える。

図4の中には、文節数=2、すなわち係り受け関係が一つしかないにもかかわらず収束までに数十ステップを必要とした例が見られる。これはネットワークの温度が高いときには比較的高い確率でエネルギーの増加方向へ状態遷移が生じるためである。ネットワークの初期温度 T_0 を小さくすればこのようなむだな状態遷移は生じにくくなるが、そのようにするとエネルギー極小状態からの脱出が困難になるため、正しい解析結果を得るためにはより一層ゆっくりと焼きなましを行わなければならない。全体としてある程度以内のステップ数でネットワークを収束させるためには、多少むだな状態遷移があってもやむを得ないと言える。

式(25)のネットワークに式(26)のパラメータを与えた場合、エネルギー最小の状態が常に文法的に正しい係り受けの組を表すとは限らない。例えば、図5に示すような文節数=3、係り受け数=3の入力があったとする。また、このときの Y_i は、

$$Y_1 = -1, Y_2 = 0, Y_3 = -1 \quad (29)$$

で与えられるものとする。このときエネルギー最小となるのは

$$u_1 = 1, u_2 = 0, u_3 = 1 \quad (30)$$

という状態であるが、これは文法的に正しい係り受けの組ではない。

にもかかわらず、図4の実験結果においては文法的に誤った解析結果となったものは一つもなかった。これは、学習用例文と解析用入力文が同一分野であるために、尤度の高い係り受けの組合せを選択することがそのまま文法的にも正しい組合せを選択することにつながるためと考えられる。また予備実験の結果得られた定数(式(26))を見ると、係り受けの尤度を表す関

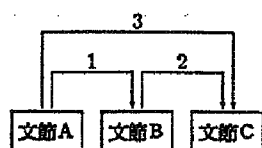


図5 文節間係り受け関係の例(2)

Fig. 5 An example of modification relation of a sentence (2).

数の係数である c が、文法的正当性を表す関数の係数である a および b よりも大きくなっているが、これも同じ理由と考えられる。

単語間係り受け共起頻度をまったく考慮しなかった場合、すなわち式(26)の c を0に変更し、

$$a=6, b=5, c=0, d=1. \quad (31)$$

とした場合の解析成功率は44%に留まった。誤った解析結果のほとんどでは、位置的にもっとも近い文節が係り先として選ばれていた。これは、単語間係り受け共起頻度という意味的な制約がなくなり、位置的に近い係り受けほど成立しやすいという運用論的な制約のみが残っている状態から容易に予測される結果と一致する。なお、この場合も文法的に誤った解析結果は得られなかったが、これは文法的正当性を表す関数の係数 a および b が、係り受けの距離を反映する関数の係数 d に比して大きいためと考えられる。

5. おわりに

相互結合型のニューラルネットを用いて、日本語の形態素解析および係り受け解析を行った。係り受け解析では単語間の意味的整合性を表す関数が解析結果に大きな影響を与えた。どちらの解析においてもパラメータを調整することによって約95%の成功率を得ることができた。入力文の長さが増加しても、ネットワークが収束するまでのステップ数はあまり増加しなかった。

ニューラルネットを利用した自然言語解析システムは、従来のシステムに比べてハードウェア化が容易であると思われる。また上に述べたように、入力文の長さが長くなっても、解析に要する時間はそれほど増加しない。したがって特定の分野における比較的複雑な文を専門に解析する自然言語解析エンジン等としての利用が考えられる。分野を限定すれば単語間の意味的關係を表す関数も比較的容易に作成できよう。

今後の課題としては、解析率の向上、形態素解析ネットワークと係り受け解析ネットワークの統合、一般のCFGを解析するネットワークの作成等が残されている。また、エネルギー関数の定数決定に Boltzmann Machine の学習アルゴリズムを適用することの可能性についても考察を進める予定である。

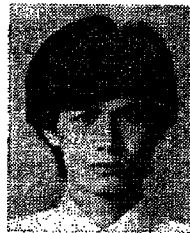
参考文献

- 1) 麻生英樹：ニューラルネットワーク情報処理，産業図書(1988)。
- 2) Hopfield, J.J.: Neurons with Graded Re-

- sponse Have Collective Computational Properties like those of Two-state Neurons, *Proceedings of the National Academy of Science USA* 81, pp. 3088-3092 (1984).
- 3) Hopfield, J. J. and Tank, D. W.: Neural Computation of Decisions in Optimization Problems, *Biological Cybernetics*, Vol. 52, pp. 141-152 (1985).
 - 4) 木村和広, 鈴岡 節, 伊藤悦雄, 天野真家: 神経回路網の連想機能を用いたかな漢字変換システムーニューロワープロの実験試作ー, 第4回人工知能学会全国大会, 9-3, pp. 301-304 (1990).
 - 5) 森 辰則, 中川裕志: Connectionist Model による構文解析モデル, 情報処理学会論文誌, Vol. 30, No. 4, pp. 447-456 (1989).
 - 6) 村瀬 功, 中川聖一: ボルツマンマシンによる文節ラティスの係り受け解析, 第38回情報処理学会全国大会論文集, 5 E-8, pp. 378-379 (1989).
 - 7) 尾関和彦: 最適文節列を選択するための多段決定アルゴリズム, 電子通信学会技術研究報告, SP 86-32, pp. 41-48 (1986).
 - 8) 奥村明俊, 山端 潔, 村木一至: ニューラルネットワークによる日本語係り受け構造の学習, 第4回人工知能学会全国大会, 11-5, pp. 353-356 (1990).
 - 9) 田村 淳, 安西祐一郎: Connectionist Model を用いた自然言語処理システム, 情報処理学会論文誌, Vol. 28, No. 2, pp. 202-210 (1987).
 - 10) 高橋直人, 板橋秀一: 相互結合型ニューラルネットワークによる日本語の係り受け解析, 第40回情報処理学会全国大会論文集, 4 F-7, pp. 464-465 (1990).
 - 11) Takahashi, N. and Itahashi, S.: Japanese Sentence Analysis Utilizing Mutually Connected Neural Network, *Proceedings of PRICAI '90*, pp. 257-262 (1990).
 - 12) 高橋直人, 板橋秀一: ニューラルネットによる日本語形態素・係り受け解析, 情報処理学会研究会報告, 90-NL-80 (1990).
 - 13) 高橋直人, 板橋秀一, 平井有三: 日本語形態素解析用ニューラルネットワークについて, 第42回情報処理学会全国大会論文集, 1 C-2 (1991).
 - 14) Waltz, D. L. and Pollack, J. B.: Massively Parallel Parsing: A Strongly Interactive Model of Natural Language Interpretation, *Cognitive Science*, Vol. 9, pp. 51-74 (1985).
 - 15) 金田一京助, 金田一春彦, 見坊豪紀, 柴田 武, 山田忠雄(編): 新明解国語辞典第二版 (磁気テープ版), 三省堂 (1974).

(平成3年2月18日受付)

(平成3年6月13日採録)



高橋 直人 (正会員)

1987年筑波大学第三学群情報学類卒業。同年筑波大学大学院博士課程工学研究科に入学, 現在に至る。自然言語処理に関する研究に従事。人工知能学会会員。



板橋 秀一 (正会員)

昭和39年東北大学工学部通信工学科卒業。昭和45年同大学院(博)電気及通信工学専攻退学。同年東北大学電気通信研究所助手。昭和47年電子技術総合研究所入所。昭和49年同所主任研究官。昭和52年ストックホルム王立工科大学客員研究員。昭和57年筑波大学電子・情報工学系助教授。現在同教授。工学博士。音声・画像・自然言語処理の研究に従事。電子情報通信学会, 日本音響学会, 人工知能学会, 日本認知科学会, IEEE, アメリカ音響学会各会員。

CONCEPT DESCRIPTION FOR SYNTHETIC SPEECH OUTPUT SYSTEM

Yoichi Yamashita*, Naoki Mizutani** and Riichiro Mizoguchi*

* The Institute of Scientific and Industrial Research, Osaka University,
8-1, Mihogaoka, Ibaraki-city, Osaka, 567 Japan.

** SHARP corporation,
2613-1, Ichinomoto, Tenri, Japan.

ABSTRACT

This paper describes a concept description scheme for speech synthesis. It is input to the synthetic speech output interface connected to various performance systems, and used for direct derivation of prosodic parameters. The concept description is composed of atomic symbols, templates and operators represented in terms of appropriate abstract level of constructs and makes it easy to generate both sentences and prosodic parameters. There are two built-in mechanisms in the templates for directly controlling the prosodic parameters. The first one is the pause marker which is generated along with words in the sentence generation. The pause marker is used to insert pauses and to locate boundaries of phrase component of pitch. The second one is the Prosody Modification Functions (PMF) embedded in the custom templates. PMF controls the the prosodic parameters for the prepared sentence pattern.

1. INTRODUCTION

Speech synthesis from concept descriptions is a very important technique for the synthetic speech output system and has several advantages over speech synthesis from text [Young,1979]. One of the goals of speech synthesis research is to construct the universal Synthetic Speech Interface (SSI) as the frontend of various Intelligent Computer Systems (ICS). To output a speech message to the user, the ICS has to send a sort of description of the message to SSI. The description can be written in several forms: texts, descriptions using Conceptual Dependency theory [Schanck,1975], and so on. The authors proposed a description scheme like a parsing tree based on the case structure, which is referred to as the concept description. The concept description is designed as the input of the SSI from the ICS. In general, text is not a best description because it is not easy to extract the information necessary for the prosody control from it. The proposed concept description can make it easy to control prosodic patterns and avoid the complexity of the sentence generation which the abstract descriptions require in SSI.

2. CONCEPT DESCRIPTION

The SSI receives the concept description as the output request from the ICS. The SSI generates a sentence and prosodic parameters and the synthesizer produces the speech sound. The concept description is composed of seven elements which are shown in the following:

(1) Atomic symbol: Verbs, nouns and adjectives which appear in a generated sentence are described by themselves as the atomic symbols.

(2) Modification template: The modification template, \$modify(A,B,OPs), describes the modification relation between the two arguments, A and B, which are constructs of sentences such as nouns, adjectives and phrases. This template contains different patterns dependent on the first argument, A, as shown in Fig.1. In sentence generation, the template generates words using the output word list prepared for each pattern which is denoted in brackets in Fig.1. The symbol 'p' is a pause marker which is generated along with words and used later in prosody control. The third

argument, OPs, is a list of prosody operators mentioned later.

(3) Case template: The case templates are prepared to describe the case information of verbs and appear as the arguments of verbs in the concept description. These templates produce function words in the sentence generation, as shown in Fig.2, because the function words express the case in Japanese sentences.

(4) Mood operator: The mood operators are prepared to describe the tense, aspect and modality information of verbs. They also appear as the arguments of verbs in the concept description. The mood information is expressed by the conjugation of the verb and the following function words in Japanese. The mood operator changes the verb into the appropriate form according to the pre-stored pattern. Fig.3 shows some examples of the mood operators.

(5) Conjunction template: The concept descriptions are processed sentence by sentence. The conjunction templates describe the relation to the preceding sentence. This template requires a sentence as its argument, and generates the conjunction before the sentence in the sentence generation, as shown in Fig.4.

(6) Custom template: The custom templates describe the frequently used expressions and the difficult expressions to describe based on the case. In the custom template, PMF can locally change the prosodic parameters. Fig.5 shows some examples of the custom templates. \$mod_acc in \$reason template is an example of PMF. Thus, PMFs facilitate the control of prosody for these expressions. It is discussed in the next section in detail.

(7) Prosody operator: The prosody operators indicate the prosodic features and are used as arguments of verbs or templates mentioned above. Three kinds of prosody operators are defined as mentioned in 4-3.

Examples of the concept description are shown in Fig.6.

3. SENTENCE GENERATION

Sentences are generated according to template matching and case grammar. The case information for each verb is output in a pre-determined order. Fig.7 shows sentences generated from the examples in Fig.6. The template has the list of output words which includes pause markers, 'p' and 'p2'. Pause markers are generated in sentences as shown in Fig.6, and are utilized to both insert pauses and boundaries of prosodic phrases in the prosody control.

4. PROSODY CONTROL

4-1. Pause Insertion

Pause insertion is necessary to the naturality of synthetic speech because the human utterance duration in a breath is limited. Two kinds of pauses are used: PS2 is a long pause (300ms) and PS1 is a short one(100ms). Pauses are inserted according to the pause markers embedded in generated sentences. The 'p2' indicates the insertion of PS2 and the 'p' indicates the potential insertion of PS1. The pause marker 'p2' is placed at the word boundary in the output word list of

s1: 地理の p 勉強を p 始めましょう (Let's begin the study of the geography.)
s2: 米が p 寒い所で p 生育しますか (Does the rice glow in the cold place?)
s3: ソ連は p 寒い国ですから p2 米が p 生育しません (The rice does not glow, because the Soviets is a cold country.)
s4: なぜ p2 米が p イランで p 生育するのですか (Why does the rice glow in Iran?)

Fig.7 Generated Sentences.

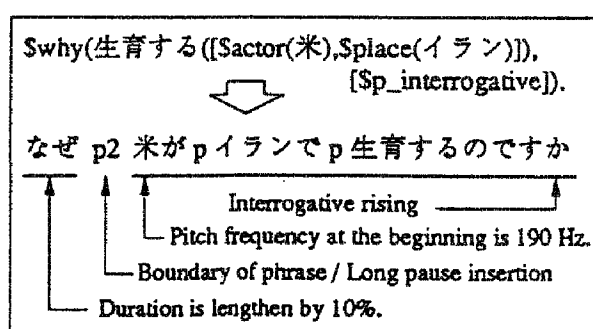


Fig.8 An Example of Prosody Control.

the templates, shown in Figs.4 and 5, where a long pause would always appear when the sentence is uttered by the human.

Some of the 'p's are replaced by PS1 taking account of number of the morae between the pauses, while every 'p2' is always replaced by PS2. If the number of morae in a one-breath phrase is over the threshold(=25), one of 'p' in the phrase is replaced by PS1 to divide the phrase into two one-breath phrases. In sentence generation, templates generate words and pause markers and results in the nest structure. A 'p' from the most inside template is selected, when more than one 'p' are found in the phrase to be divided. The replacement of pause markers is recursively carried out.

4-2. Boundary of Phrase Component of Pitch

The pitch contour is controlled based on the model of the addition of the lexical word accent pattern to the declination line, and the phrase component of pitch is represented by the declination line. The boundary of the phrase component is determined in the same way as the pause insertion except using the smaller mora threshold (=15). Thus, the pause always makes the boundary of the phrase component of pitch.

4-3. Prosody Operators

The prosody operators described in the concept description can adjust prosodic parameters. Three prosody operators are currently defined: \$p_prominence(A), \$p_interrogative and \$p_speed(A). \$p_prominence increases the accent component of pitch and prosodically emphasizes its argument A. \$p_interrogative requires that the pitch rises at the end of the sentence. \$p_speed(A) controls the utterance speed of its argument A, partially in the sentence.

4-4. Prosody Modification Function (PMF)

The Prosody Modification Functions (PMF) are used only in the custom template while the prosodic operators are described in the concept description. Three PMFs, \$mod_dur, \$mod_bpit and \$mod_acc, are prepared to control the duration, the pitch frequency at the beginning point of phrase component and the accent prominence component, respectively.

4-5. An Example of Prosody Control

Fig.8 shows an example of prosody control. Four modifications are carried out in this example. Both the duration for first word and the pitch frequency at the beginning of second phrase are derived from the PMFs in the custom template \$why(A,OPs). The boundary of the phrase and the long pause insertion are determined by the use of the pause marker. And, the prosody operator, \$_interrogative, in the concept description gives the pitch rising at the end of the sentence.

5. CONCLUSION

This paper proposed the concept description for the Synthetic Speech Interface (SSI) and a method of the direct control of the prosodic parameters for the synthetic speech using the concept description. The idea that prosody markers and PMFs control the prosody is useful for any language, though templates are dependent on the language. Our system of the synthetic speech output from the concept description has been implemented in C-Prolog, adopting a CAI system as the Intelligent Computer System.

REFERENCE

- Schank R.C. (1975), "Conceptual information processing", 3, 22-82, North-Holland publishing company.
- Young S.J. & Fallside F. (1979), "Speech synthesis from concept: A method for speech output from information systems", J.A.S.A., 66, 3, 685-695.

MASCOTS: Dialog Management System for Speech Understanding System

Tetsuya YAMAMOTO[†], Yoshikazu OHTA[†], *Nonmembers*, Yoichi YAMASHITA^{††},
Osamu KAKUSHO^{†††} and Riichiro MIZOGUCHI^{††}, *Members*

SUMMARY This paper describes a dialog management system called MASCOTS which manages a dialog between a user and a problem solving system through spoken Japanese and helps the speech understanding system in its language processing. MASCOTS tries to predict the next user utterance based on the architecture for managing dialog with two stacks and plan information. MASCOTS not only contributes to making language processing efficient, but also works for a problem solving system. MASCOTS identifies the kind of the utterance and standardizes its representation form in place of a problem solving system. In this paper, the architecture of MASCOTS is discussed focusing on the characteristics of dialog and two ways of predicting the next user utterance exchanging the information with the language processing system.

1. Introduction

There are two major types of studies about dialogs; one is to analyze the structure of dialogs⁽¹⁾ and the other is to develop a speech dialog system including the problem solving such as information retrieval and so on^{(2),(3)}.

However, our motivation for the study of dialog is different from both of them. We have been developing a speech understanding system SPURT-I (Speech Understanding system with Rule-based and Topic-directed architecture)⁽⁴⁾ so far, which accepts one-way Japanese speech such as telling a story describing simple scenes. As a next step of this research, we have been involved in the realization of a two-way communication between a user and a problem solving system through spoken Japanese. To this end, SPURT-I has to accept the user utterances included in the dialog. Unlike one-way speech, the dialog has many inherent characteristics, so it is possible to utilize them for speech understanding process. In order to manage the dialog and help the language processing subsystem ASP (Association-based Parser)⁽⁵⁾ in SPURT-I, we are currently developing a dialog management system

MASCOTS (MANagement System for CONversation using Twin-stacks and Sr-plan)⁽⁶⁾. MASCOTS manages the complex flow of dialog using two stacks and plan information and predicts the next user utterance from the context of the dialog. MASCOTS also plays a few important roles for a problem solving system as follows.

During dialog, users do not always return responses expected by the system. For instance, when the system asks a question, the user may ask another question instead of answering it. Furthermore, the representation of the user utterance takes various forms even though it might have the same meaning. Therefore MASCOTS identifies the kind of the utterance and standardizes its representation form before sending the recognized utterance to the problem solving system.

In the following sections, we discuss the architecture of MASCOTS focusing on the characteristics of dialog and exchanges of the information with ASP.

2. Basic Concept

2.1 Configuration of the Total System

Figure 1 shows the block diagram of the total system configuration. Our speech I/O interface is composed of a speech understanding system SPURT-I, a speech synthesizer and MASCOTS. SPURT-I translates user's speech into text in such ways that (1) acoustic processing subsystem named SPREX (Speech Recognition EXpert)⁽⁷⁾ converts user's utterance to a sequence of phonemes, (2) word hypothesizer BCG (Bunsetsu Candidate Generator) transforms the phoneme sequence into a set of conflicting candidates in every syntactic unit ("bunsetsu"), and (3) ASP identifies a correct sequence of "bunsetsu"s in the candidate lattice. On the other hand, the speech synthesizer generates speech from the outputs of the problem solving system and utters it to the user. MASCOTS locates between these speech processing systems and the problem solving system, and manages the utterances exchanged by the user and the problem solving system, that is a dialog. And processing in MASCOTS is tightly connected to ASP, so these two systems share some parts of their knowledge bases.

Manuscript received January 10, 1991.

Manuscript revised April 8, 1991.

[†] The authors are with the Faculty of Engineering, Kansai University, Suita-shi, 564 Japan.

^{††} The authors are with the Institute of Scientific and Industrial Research, Osaka University, Ibaraki-shi, 567 Japan.

^{†††} The author is with the Faculty of Science and Technology, Ryukoku University, Otsu-shi, 520-21 Japan.

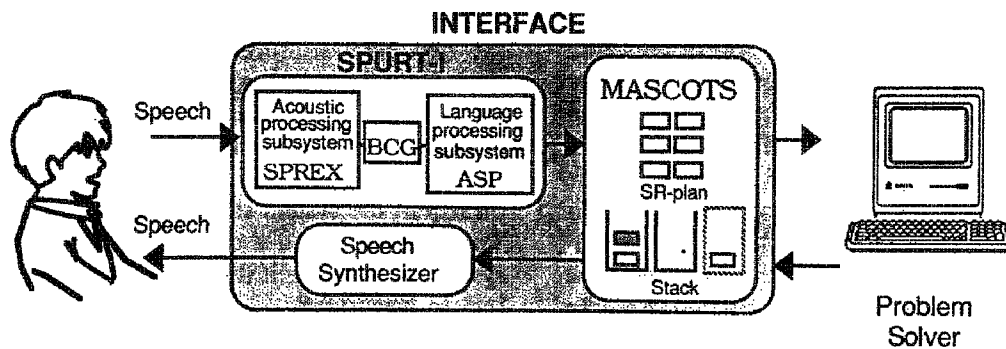


Fig. 1 Total system configuration.

This total interface is designed independently of the type of problem solving system. In other words, it is independent of the domain. And appropriate functions for speech communication are assigned to each module.

2.2 Characteristics of Dialog

Dialogs are classified into some types from various points of view. MASCOTS deals only with what we call goal-oriented dialogs⁽⁸⁾ such as ones appearing in consultation, information retrieval, CAI and so on. In such dialogs, an utterance, which of course does not always consist of only one sentence, contains at least either of a stimulus (requirement, question, order and so on) to the opponent or a response to the stimulus given by the opponent explicitly or implicitly. Communication is thus performed by giving stimuli or responses to each other and some additional information such as confirmation, explanation, condition to restrict the contents and so on. According to close investigation of dialogs, we get seven characteristics inherent to the dialog.

1. There exists one-to-one correspondence between the stimulus and the response.
2. Basically, a response is given immediately after a stimulus.
3. The last interaction completes first.
4. If more information is needed to respond, the user asks for related information.
5. When the premise of the system stimulus does not hold, the user confirms it.
6. When the user cannot understand the intention of the system stimulus, he/she asks about it.
7. When the user failed to hear the system, he/she asks to repeat it.

2.3 SR-Plan and Twin-Stacks

As mentioned in the previous section, we believe that the stimulus and the response are basic components of the goal-oriented dialogs. So, to grasp the structure of the dialog, we introduce the concept of

Stimulus-Response plan (SR-plan for short) which helps MASCOTS interpret the interaction between the system and the user. MASCOTS has the user SR-plans and the system SR-plans which are for dealing with the interactions constructed by user stimuli and system responses and by system stimuli and user responses, respectively.

For manipulating SR-plans, MASCOTS employs a stack architecture. When the system provides a stimulus with the user, the corresponding system SR-plan is pushed down into the stack for the system SR-plan called "system stack". Similarly, the other stack "user stack" is used when the user provides a stimulus, that is the case which the user SR-plan is activated. Only the topmost plan in these two stacks, which we call current plan, is active and the processing of the other plans are suspended. When the current plan gets its response and it is confirmed explicitly or implicitly, MASCOTS pops up that plan and stores it in the history data base. In ordinary cases, a response corresponds to the latest stimulus independently of the history of the other stimuli and responses. This shows the adequacy of the stack structure. However, there may be some cases in which this structure is no valid. The process in such a case will be discussed in Sect. 2.5. Litman⁽⁹⁾ uses one stack for dealing with a dialog. In our case, however, since the procedures of two kinds of SR-plans are completely different, two kinds of stacks are used to make the implementation easier. Furthermore, the movement of a current plan on twin-stacks indicates the movement of initiative of the dialog. This feature may also help build two speakers' models in the future works.

2.4 Procedure of SR-Plan

The following is a basic procedure of system SR-plan.

- Step 1 Activate and push down SR-plan into the system stack when the system provides a stimulus.
- Step 2 Send the content of the utterance to the utterance synthesizer.
- Step 3 Extract the keywords from system utterance

and send them to ASP.

Step 4 According to the keywords found in input lattice by ASP, predict the possible types of responses and send one to ASP.

Step 5 When ASP analyzes the user utterance as a response, standardize the result and send it to the problem solving system. When it is not a response, this plan is suspended.

Step 6 When the system confirms the user response explicitly or implicitly, this plan is popped up from the system stack.

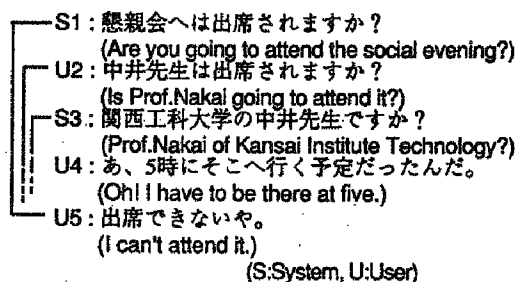
Details of the keyword and the way of prediction will be discussed in Sect. 3. The procedure of user SR-plan is omitted except the description of the special case as follows.

In the case that the user stimulus does not need the system judgment, MASCOTS does respond to the user in place of the problem solving system without sending the user utterance to the system but reporting the occurrence of it. This is seen, for instance, when the user asks what the system said.

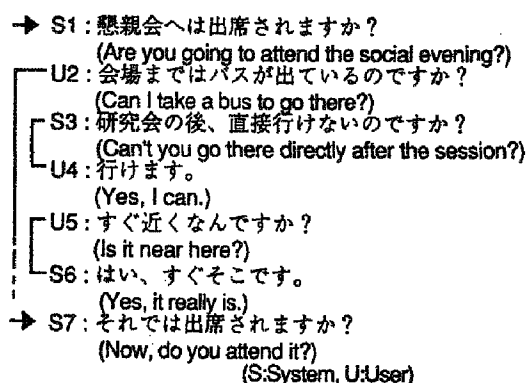
2.5 Extension of the Stack Manipulation

Our twin-stacks work well in ordinary dialog. However, the following cases require special procedures since they are beyond the basic stack architecture.

(1) When the user responds to lower stimulus than topmost stimulus in the stack.



(a) An example of type (1).



(b) An example of type (2).

Fig. 2 Examples of special patterns.

(2) When the system or the user omits a response to the opponent stimulus in the case that the opponent poses inappropriate stimulus for his/her thinking.

Figure 2(a) and (b) show examples of types (1) and (2), respectively.

In Fig. 2(a), U5 corresponds not to S3 which is on top of the system stack but to S1 locating under S3. To cope with such a case, MASCOTS goes down the system stack to see if there exists a stimulus corresponding to this utterance if and only if the following conditions both hold; (i) the utterance cannot be recognized as either a response to the topmost system stimulus or a new stimulus and (ii) there exist suspended SR-plans in the system stack. When such a stimulus is found, all the SR-plans higher than this stimulus are popped up from the two stacks.

The situation of type (2) is rather difficult to detect. If the system or the user mentions the inadequacy of the opponent stimulus clearly, all MASCOTS has to do is to pop up the corresponding SR-plan. For instance, in Fig. 2(b), if the system says “出てません” (No, you can't.) after U2, MASCOTS can pop up the SR-plan. However, as S3 in Fig. 2(b), if he/she asks the other stimulus without saying anything about the inadequacy, MASCOTS cannot see whether it is adequate or not. So, only in such a situation, MASCOTS checks whether the next stimulus after some interactions (S7) is same as the suspended stimulus (S1) since if he/she does not respond to the inadequate stimulus and asks the other stimulus, he/she will return the previous stimulus after some interactions are completed. When the same stimulus is found, all the SR-plans higher than one corresponding to the inadequate stimulus are popped up together with itself.

3. Prediction of the Next User Utterance

3.1 Top-down Processing

Unlike one way speech such as telling a story, the user utterances in the dialog can often be predicted since most of them are responses to the stimuli of the system. Figure 3 shows the order in which the user utterance is expected.

Furthermore, the forms of responses and stimuli

- After system stimulus
(Current plan is in the system stack)
 1. Response to it
 2. New stimulus
- After system response to user stimulus
(Current plan is in the user stack)
 1. Its confirmation
 2. Response to the suspended stimulus of the system
 3. New stimulus

Fig. 3 Expectation of user utterances.

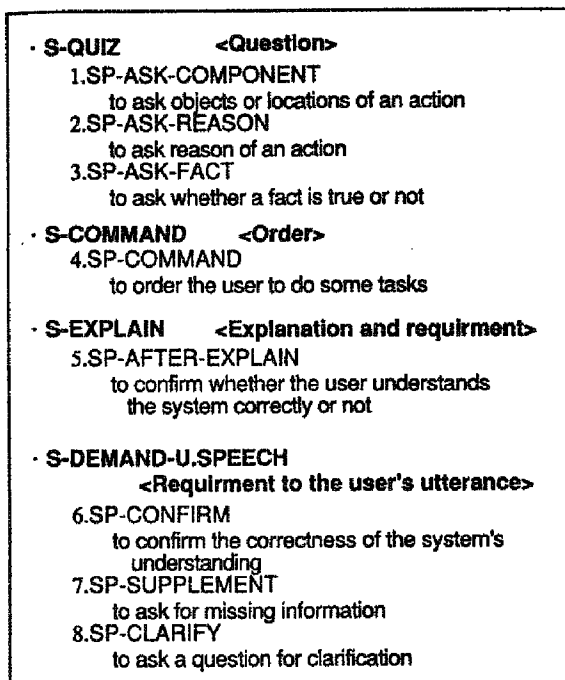


Fig. 4 System SR-plans.

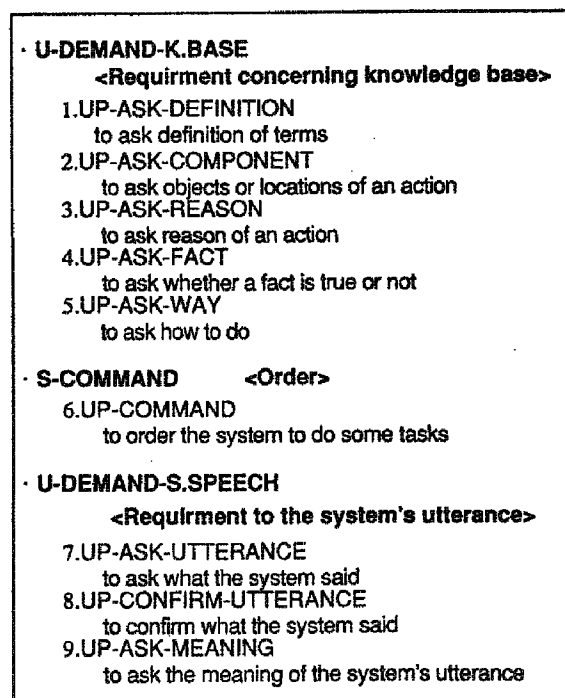


Fig. 5 User SR-plans.

depend on the type of interaction. To predict the utterance form, we classified SR-plans into 17 categories according to the type of interaction. Figures 4 and 5 show examples of system SR-plans and user SR-plans, respectively.

Each system SR-plan has response templates corresponding to the plausible utterance forms of the user responses in its interaction. Similarly, each user SR-plan has stimulus templates for the user stimuli. For

System : お車は正しい位置に駐車しましたか?
(Did you park your car at the right place?)

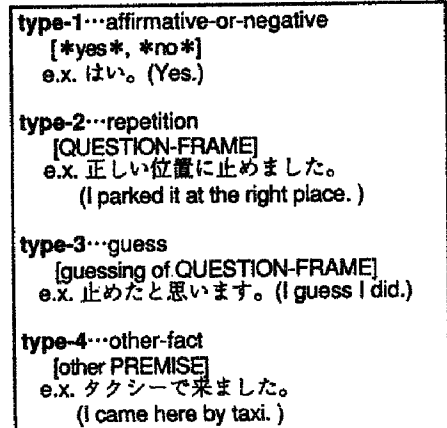


Fig. 6 Response templates in SP-ASK-FACT.

instance, SP-ASK-FACT, which is one of the system SR-plans and is for dealing with yes-no-question, has affirmative-or-negative, repetition, guess and other-fact types of response templates as shown in Fig. 6.

However, the contents of these templates depend on the contents of the previous system utterance except ones independent of the contents such as affirmative-or-negative type of template. So, each template is constructed by two kinds of generic keywords. One is registered in advance. For instance, a generic keyword *yes* includes the keywords of “はい”, “うん”, etc. (“yes”, “yeah”, etc.). The other is instantiated when the stimulus of the system is given together with some additional information such as the premise and the intention of the stimulus. For instance, the generic keyword QUESTION-FRAME will be instantiated to include the words used in the system utterance. In type-2 in Fig. 6, QUESTION-FRAME will include the keywords of “車” (“car”), “正しい” (“right”), “位置” (“place”) and “駐車する” (“park”) after the system asked a question that “お車は正しい位置に駐車しましたか?” (“Did you park your car at the right place?”). This generic keyword will also include the keywords defined under the same category in thesaurus, which are “自動車”, “乗用車”, “適切な”, “止める” and so on (“automobile”, “motorcar”, “appropriate”, “stop” and so on). This means that the templates written in an SR-plan are instantiated when the latter kind of generic keywords are instantiated according to the last system utterance.

On the other hand, as for the prediction of the user stimulus, we take the other knowledge into consideration in addition to those processings. We think that there are four patterns in which the user returns stimulus instead of response.

1. The user asks for related information in order to respond the system stimulus.
2. The user confirms the premise of the system stimulus.

lus when it does not hold.

3. The user asks the intention of the system stimulus when he/she cannot understand it.

4. The user asks what the system said when he/she failed to hear the system.

So, MASCOTS generates stimulus templates based on the contents, the premise, the intention of the last system stimulus and this knowledge. But, we have not considered about the use of the intention yet.

3.2 Bottom-up Processing

Top-down processing has at least two shortcomings. One is that there is no ordering among response templates or among stimulus templates. The other is that too many templates are generated. In order to overcome these difficulties, we introduce bottom-up analysis. When a word lattice is given to ASP, ASP scans all the candidate words to find out some keywords which are elements constructing the templates. Then MASCOTS selects all the templates composed of only the keywords found by ASP. These templates will be ordered according to the top-down expectation and some other heuristics described below.

3.3 Ordering of Templates

The selected templates are scored by four kinds of heuristics as follows.

1. according to the top-down expectation (Top-Down point)
2. according to recognition score in acoustic processing for each keyword (Recognition point)
3. according to the kind of keywords (Key-Word point)
4. according to frequency of the keywords in the history of the dialog (Context point).

The template which gets the highest score is sent to ASP. The success of the ASP analysis using this template indicates that the utterance is of type directly inferred from the template. When the analysis fails, the next probable template is sent and the process is repeated until it succeeds. If all the expectations fail, then ASP performs using only the dependencies between the words without any information from MASCOTS.

```

S1: 懇親会へは出席されますか?
    (Are you going to attend the social evening?)
U2: 中井先生は出席されますか?
    (Is Prof. Nakai going to attend it?)
S3: はい、出席されます。
    (Yes, he is.)
      (S: System, U: User)
  
```

Fig. 7 An example of dialog.

4. Example of the System Behavior

Figure 7 shows an example of a dialog at a registration desk.

First, the problem solving system, in this case the registration desk system, asks a question if the user will attend the social evening or not. This is what we call a stimulus asking a fact, so system SR-plan SP-ASK-FACT is activated and pushed down into the system stack (Fig. 8(a)). Then the generic keyword QUESTION-FRAME is instantiated from the system

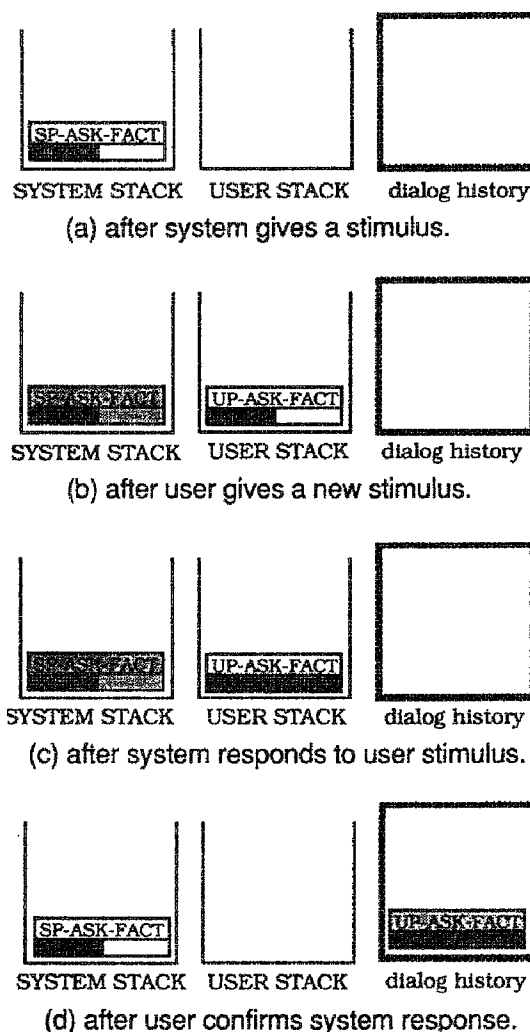


Fig. 8 Illustration of twin-stacks.

```

(0.9476 moesasenaiwa)      (0.9441 syuppatusimasuka)
(0.9385 nakasenusouda)     (0.9342 syuppatusimaiga)
(0.9384 nakasemasuwa)     (0.9285 syussekiimasuka)
(0.9369 nakaisenseiha)     (0.9219 hissorimasuga)
(0.9364 nakaisenseiga)     (0.9204 kyuusensimasuka)
:                             :
:                             :
  
```

Fig. 9 Input lattice for U2.

utterance as mentioned in Sect. 3.1. Simultaneously, the templates of the expected responses in SP-ASK-FACT are instantiated using this generic keyword. In this case, templates of affirmative, negative, assertion of a fact and guess of a fact are expected. After that, the content of utterance is sent to the speech synthesizer and S1 is uttered.

Next, U2 is uttered by the user. The input of ASP is actually a word lattice, so the lattice includes not only correct candidate words but also many incorrect ones (Fig. 9). ASP scans all the candidate words and picks up the generic keywords QUESTION-FRAME (including konsinkai, syusseki suru, etc.), *do?* (including -ka, -ne, etc.), *fact* (including syusseki suru, hazimaru, etc.) and 2 other generic keywords from them in this case. MASCOTS tries to find the templates which consist of only these generic keywords from the response templates in SP-ASK-FACT (current plan) and the stimulus templates in all user SR-plans. MASCOTS selects four templates in this case; <RESPONSE7>, <UP-ASK-FACT3> and 2 others. <RESPONSE7> is a response template in the current plan SP-ASK-FACT and consists of QUESTION-FRAME. <UP-ASK-FACT3> is a stimulus template in the user SR-plan UP-ASK-FACT and

consists of *fact* and *do?*. Bottom-up processing is shown in Fig. 10. Next, these selected templates are scored using four kinds of heuristics. Since U2 is uttered after the system stimulus, top-down inference gives higher T-D-point to the response templates than to the stimulus templates. Other kinds of points are given in similar manner. The scoring results of the above two templates are shown in Fig. 11.

After all, the stimulus template <UP-ASK-FACT3> gets the highest point and is sent to ASP. ASP analyzes U2 with this template successfully in this case. And U2 is recognized as a new stimulus and the user SR-plan UP-ASK-FACT is activated. This plan is pushed down into the user stack and the previous plan SP-ASK-FACT is suspended (Fig. 8(b)). Then the kind (result of recognition, that is stimulus) and the content of utterance which is extracted using information in the template are sent to the problem solving system. However, if ASP cannot analyze the input with the first template successfully, the second template is sent to ASK and this process is repeated until the analysis succeeds.

After S3 answers to U2, the current plan (corresponding to the interaction by U2 and S3) waits a confirmation from the user (Fig. 8(c)). Later, if MASCOTS judges that the user confirms it explicitly or implicitly, then the current plan is popped up from the user stack. Then the suspended plan SP-ASK-FACT becomes active again and waits a response from the user (Fig. 8(d)).

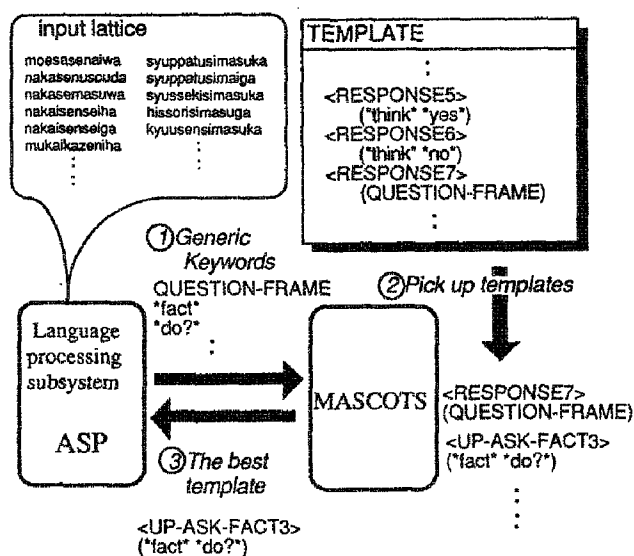


Fig. 10 Bottom-up processing.

RESPONSE7	T-D-point	10
	Rec-point	1.7
	K-W-point	0
	Con-point	0
	total	11.7
UP-ASK-FACT3	T-D-point	6
	Rec-point	3.2
	K-W-point	3
	Con-point	0
	total	12.2

Fig. 11 Scoring results of templates.

- S1: 恐れ入りますが、お名前は何とおっしゃいますか?
(May I have your name, please?)
 - U2: 山田といいます。
(My name is Yamada.)
 - S3: どちらに勤めていらっしゃいますか?
(What is the name of your company?)
 - U4: 大阪電器です。
(Osaka Electronics Corporation is.)
 - S5: 懇親会へは出席されますか?
(Are you going to attend the social evening?)
 - U6: 懇親会は何時から始まりますか?
(What time does it begin?)
 - S7: 5時から7時までの予定です。
(From five to seven.)
 - U8: あ、そうですか?
(I see.)
 - U9: 中井先生は出席されますか?
(Is Prof. Nakai going to attend it?)
 - S10: 関西工科大学の中井先生ですか?
(Prof. Nakai of Kansai Institute Technology?)
 - U11: そうです。
(Yes.)
 - S12: 研究会、懇親会とも出席されます。
(He will attend both the technical session and the social evening.)
 - U13: それでは出席します。
(Then, I will attend it.)
- (S: System, U: User)

Fig. 12 Dialog used in the experiment.

Table 1 Performance evaluation.
(1 male adult, a 400-word vocabulary)

	Kind of sentence	Number of selected templates	Order of correct template	Result
U2	Res.	4	2	○
U4	Res.	10	3	○
U6	Sti.	8	1	△
U8	Conf.	6	1	△
U9	Sti.	3	1	△
U11	Res.	2	2	○
U13	Res.	4	3	X
Average		5.3	1.9	6/7

5. Performance Evaluation

We made an experiment to evaluate the performance of MASCOTS with a 400-word vocabulary. In the experiment, seven user utterances are picked up from the conversation shown in Fig. 12 between one male adult and the registration desk system. Each sentence consists of 1.9 syntactic units on the average. As for ASP input, the average number and the maximum number of candidates in a unit are 37 and 97, respectively. And the average rank of correct candidates is 11th. Table 1 shows the result of the evaluation. This shows that MASCOTS selected 5.3 templates on the average from 79 templates in all and the ordering mechanism put the correct templates at 1.9th rank on the average. The circle in "Result" means the perfect success of word identification in the sentence, the triangle means the success of meaning analysis including mismatch of the function words, say "to me" for "for me", and the saltire means the failure. The result of this experiment shows that MASCOTS helps ASP process the user utterances in the dialog effectively. ASP tends to depend not on the dependencies between the words but on the recognition score since the short sentences are often used in the dialog. In the extreme case, if the user utters only one word, ASP cannot analyze it successfully without the information from MASCOTS.

6. Conclusions

We have discussed a dialog management system MASCOTS which helps a speech understanding system. The basic mechanism of MASCOTS is based on the SR-plan and twin-stack architecture. And we also discussed about the way of predicting the next utterance by top-down and bottom-up processing. Currently, the prototype of MASCOTS is being implemented in Common Lisp and Flavors on Symbolics 3620.

Unlike other dialog systems developed so far, MASCOTS is a part of I/O interface for realization of a two-way communication between a user and a problem solving system through spoken Japanese. So, MASCOTS is independent of the domain for which

the problem solving system performs. Furthermore, MASCOTS identifies the kinds of user utterances and standardizes representation forms of them using syntactic information in utterance templates to help the problem solving system. And in place of the problem solving system, MASCOTS responds to the user stimulus which is inherent to the dialog, for instance, in the case that the user asks what the system said. These characteristics suggest that the developers of the problem solving systems do not have to care anything about dialogs. Neither do the developers of the speech understanding systems, of course.

When the speech understanding system tries to understand the user speech in the dialog, it can utilize top-down information sent from MASCOTS in addition to bottom-up information found in the input lattice. MASCOTS undertakes everything about dialogs to help other systems. So, MASCOTS employs a lot of general knowledge concerned with dialogs. And next user utterance is dynamically predicted using that knowledge together with information given by the problem solving system such as system utterance, the premise of the stimulus, the intention of the stimulus and so on. In other words, MASCOTS works merging fragments of domain knowledge into general knowledge about dialogs.

As for the ways of prediction, top-down and bottom-up processings make up for their shortcomings each other. Only by top-down processing, there is no ordering among response templates or among stimulus templates and too many templates are generated. Conversely, only by bottom-up processing, characteristics of dialog cannot be utilized for the prediction mechanism effectively.

Currently, we are investigating the use of a dialog history. To this end, at least, it is necessary to build both speakers' models and to track the expansion of dialog.

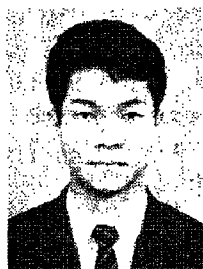
Acknowledgements

This work is partly supported by a Grant-in Aid for Scientific Research on Priority Areas "Advanced Man-Machine Interface Through Spoken Language" (No. 01608006) from the Ministry of Education.

References

- (1) Doi A., Kashioka H., Dan S., Hirai M., Abe N. and Kitahashi T.: "Dialogue analysis by constraint-learning", Proc. of the third symposium on advanced man-machine interface through spoken language (1989).
- (2) Niimi Y. and Kobayashi Y.: "Topic Management and Keyword Prediction in a Speech Dialog System", IEICE Technical Report, SP87-103 (1987).
- (3) Ohsawa I.: "A Computational Model of an Intelligent Agent who Talks with a Person", IEICE Technical Report, NLC89-6 (1989).

- (4) Hori M., Tsujino K., Mizoguchi R. and Kakusho O.: "A speech understanding system SPURT-I —Dynamic clustering method and performance evaluation—", Trans. IEICE, **J72-D-II**, 8, pp. 1291-1298 (Aug. 1989).
- (5) Hori M., Mizoguchi R., Kawachi M., Uehara K., Toyoda J. and Kakusho O.: "Association-based parser for speech understanding system—Framework design based on Cognitive exploration—", Trans. IEICE, **J71-D**, 5, pp. 774-781 (1988).
- (6) Yamamoto T., Ohta Y., Yamashita Y. and Mizoguchi R.: "Dialog management system MASCOTS in speech understanding system", Proc. of ICSLP-90, pp. 1301-1304 (1990).
- (7) Tsujino K., Mizoguchi R. and Kakusho O.: "A continuous speech recognition system SPREX —A knowledge engineering approach to speech recognition", Proc. of the 3rd Western Pacific Regional Acoustics Conference, 2, pp. 771-774 (1988).
- (8) Iida H.: "Pragmatic Characteristics of Natural Spoken Dialogues and Dialogue Processing Issues", Journal of JSAI, 3, 4, pp. 49-56 (1988).
- (9) Litman D. J. and Allen J. F.: "A Plan Recognition Model for Subdialogues in Conversations", Cognitive Science, 11, pp. 163-200 (1987).



Society of Japan.

Tetsuya Yamamoto was born in Kyoto, Japan, on January 12, 1967. He received the B.E. and M.E. degrees Kansai University, Osaka, Japan, in 1989 and 1991, respectively. In 1991, he joined the Matsushita Electric Works, Ltd. His research interests are in speech understanding and dialog modeling. He is a member of Information Processing Society of Japan, Japanese Society for Artificial Intelligence and the Acoustic



of the Institute of Electrical Engineers of Japan, the Institute of Electronics, Information and Communication Engineers of Japan and Information Processing Society of Japan.

Yoshikazu Ohta was born in Osaka, Japan, on December 6, 1923. He received the B.E. and Ph.D. degrees from Osaka University, Osaka, Japan in 1948 and 1971, respectively. Since 1958, he has been engaged in research at Kansai University. Since 1970, mainly he has researched on 2-D-digital filters, identifications of system parameters and numerical calculations of electrical and magnetic field by B. E. M. He is a member



Japanese Society of Artificial Intelligence and European Speech Communication Association.

Yoichi Yamashita was born in Ehime, Japan, in 1959. He received the B. S. and M.E. from Osaka University in 1982 and 1984, respectively. He is a technical official in the Institute of Science and Industrial Research of Osaka University, Japan. His research interests include digital signal processing, speech understanding, speech synthesis and human-computer interaction. He is a member of the Acoustic Society of Japan,



was a professor at the Institute during 1974-1989. There he is currently a Professor of Electronics. His research interests include electronic measurement, pattern recognition, signal processing, speech understanding system and intelligent information processing system. Dr. Kakusho is a member of the Information Processing Society of Japan and the Acoustical Society of Japan. He received honorable mention for the Pattern Recognition Society award in 1985, and the IEICE award in 1987. Currently he is a professor emeritus at Osaka University as well as a professor at the Department of Electronics and Informatics, Ryukoku University.

Riichiro Mizoguchi was born in Tokyo, Japan, on October 13, 1948. He received the B.S., M.S. and Ph.D. degrees from Osaka University, Osaka, Japan, in 1972, 1974 and 1977, respectively. In 1977, he joined the Osaka Electro-Communication University. From 1978 to 1986 he was a research associate of the Institute of Scientific and Industrial Research, Osaka University. Since 1990, he has been a Professor there. His

research interests include nonparametric data analysis, speech understanding system, expert system and intelligent tutoring system. He received awards from the Pattern Recognition Society and the Institute for Electronics, Information and Communication Engineers of Japan in 1983 and 1987, respectively. Dr. Mizoguchi is a member of the Institute of Electronics, Information and Communication Engineers of Japan, the Information Society of Japan, Japanese Society for Artificial Intelligence, IEEE and AAAI.

Reducing Complexity of Constraint-Based Grammars*

HASIDA, Kôiti

1 Introduction

It is practically impossible to delimit the scope of information potentially relevant to the survival or benefit of a cognitive agent. On the other hand, the information-processing capacity of a cognitive agent is severely limited. Only a very small part of the relevant information can therefore actually be reflected in the actions (including inferences) by the cognitive agent at each context. This gives rise to a very complex flow of information in cognitive processes, because the information really exploited should drastically vary from one context to another, in order to encompass as much of the relevant information as possible across various contexts.

Due to this complexity, descriptive formalisms which stipulate and hence restrict information flow fail to capture subtleties of cognition. In this connection, the major advantage of constraint-based approaches, among other intertwined benefits, is that they are basically free from stipulation of (restriction on) information flow, providing a basis on which to construct scientific theories with tractable complexity.

The transformation operations employed in some linguistic theories (Chomsky 1981, 1986) are officially claimed to be free from processing order. This claim may probably be justified, but still those theories impose too much limitation on information flow, because successive transformations give birth to sequences of intermediate structures, and information flow is restricted to be along this sequence. Transformations are procedures by nature, and the claim that transformations imposes no processing directions is as vacuous as a claim that FORTRAN and LISP programs do not restrict information flow because they are regarded as constraints on the structure of the temporal sequence of working memory contents. Just as those procedural programming languages, transformational grammars postulate intermediate stages of computation, inviting otherwise excessive complexity in the resulting theories.

The linguistic theories often termed 'unification-based,' 'non-transformational,' 'information based' and so on (Bresnan 1982, Gazdar et al. 1985, Pollard and Sag 1987, Gunji 1987) provide a basis for designing grammars genuinely exempt from such a restriction on information flow and the resulting complications of theories. For instance, *slash* feature in some of those theories could be regarded as a genuine constraint-based mechanism, in the sense that it captures long distance dependencies in terms of relationships within

*In *Situation Theory and Its Applications, II*, Jon Barwise, Jean Mark Gawron, Gordon Plotkin, and Syun Tutiya, eds. 1991, pp. 405-423.

a single grammatical structure rather than in terms of procedure (relationships among many intermediate structures).

However, those constraint-based theories are not yet entirely free from the procedural line of reasoning. Still they have several aspects which involve what appear to be intermediate structures. The major purpose of this paper is to point out some of those aspects and propose genuinely constraint-based alternatives. There are mainly two of such aspects that are discussed below. One is the phrase-structure based description of the relationships among words, and the other is quantifier storage (Cooper 1983) to deal with scopings. So called mother categories, together with subcat feature, may be regarded as encoding intermediate stages of the computation to work out intrasentential relationships. Quantifier storages would be more readily recognized as intermediate structures.

In place of phrase structure and subcat feature, we will propose an alternative treatment based on what could be called *potential energy*. The system of constraint we employ below consists of a symbolic (digital) and an analog aspect. The symbolic aspect is a sort of first order logic program, and the analog aspect is potential energy, which represents the degree of dissatisfaction of constraints. A common usage of such an analog component is to provide preferences among *possible* combinations of propositions, but we consider that simply every combination is possible, and that potential energy defines analog *truth values* between 0 (absolutely inconsistent) and 1 (absolutely consistent), excluding 0 and 1. In this view, potential energy does not serve as a secondary means used when the symbolic aspect fails to determine the answer, but plays a primary role to be tightly coupled with the symbolic aspect. This results in a reduction of symbolic (combinatorial) complexity of the overall design of the constraints, parts of constraints formerly encoded symbolically being substituted with energy minimization.

The alternative approach we propose below to handle quantifier scopes will be obtained also by eliminating intermediate stages of computation encoded in linguistic structures. Potential energy will be employed here as well, though playing a somewhat minor role of determining preferred readings.

2 Constraint and Potential Energy

We assume that what the world might be like is captured in terms of constraints of various sorts, which are represented uniformly as first order logic programs of a clausal form plus some analog component. For example, a piece of constraint might look like the following clause, when we restrict our attention to the symbolic aspect.

$$(1) p(X) \leftarrow q(X,Y) \wedge r(Y).$$

Names beginning with capital letters are variables. The other names and special symbols are constants, including predicates and functors. As usual, each clause is a disjunction of atomic formulae which is universally quantified with respect to every variable it contains. So clause (1) is an abbreviation of the following first order formula.

$$(2) \forall X \{p(X) \Leftarrow \exists Y \{q(X,Y) \wedge r(Y)\}\}$$

For any binary predicates ρ_i ($1 \leq i \leq n$), an object α satisfying (3) may be written as a record-like term shown in (4).

$$(3) \rho_1(\alpha, \beta_1) \wedge \rho_2(\alpha, \beta_2) \wedge \cdots \wedge \rho_n(\alpha, \beta_n)$$

$$(4) \{\rho_1/\beta_1, \rho_2/\beta_2, \cdots, \rho_n/\beta_n\}$$

Some binary predicates, called *features*, stand for partial functions. For instance, if ρ is a feature, then $\rho(\alpha, \beta) \wedge \rho(\alpha, \gamma)$ implies $\beta = \gamma$. An atomic formula with a feature is called a *feature specification*. Although the record notation like (4) is restricted to cases where every ρ_i is a feature in usual practices (Johnson 1988), we use this notation more loosely for expository conveniences.

Clauses are regarded as identical as far as they are transformed to each other by syntactic transformations preserving the logical equivalence. For instance, one and the same clause may be represented in many different ways including the following three.

$$(5) p(f(X)) \leftarrow q(\{a/X, b/Y\}, Y) \wedge r(Y).$$

$$(6) \neg q(B, Y) \vee A \neq f(X) \leftarrow \neg p(A) \wedge r(Y) \wedge B = \{a/X, b/Y\}.$$

$$(7) p(A) \vee \neg q(B, Y) \vee \neg r(Y) \vee A \neq f(X) \vee \neg a(B, X) \vee \neg b(B, Y).$$

Information processing concerns hypotheses. A hypothesis is the negation of a clause: that is, a conjunction of atomic formulas existentially quantified in terms of all the variables appearing in it. Computation begins given an initial hypothesis, which is successively rewritten to new hypotheses by abduction (backward chaining) and factoring (unification), while checking consistency by deduction (forward chaining).¹ Abduction may be regarded as resolution in the standard sense, and factoring is also understood in the standard fashion. Deduction is mainly for consistency checking. Some factorings are obligatory, as a part of consistency checking. A typical example is the unification of two feature specifications sharing both the feature and the first argument. That is, if ρ is a feature, then $\rho(\alpha, \beta)$ and $\rho(\alpha, \gamma)$ must unify, so that β and γ unify.

The atomic formulas and literals are classified into two categories, *defined* and *free*. Defined atomic formulas and defined literals are those with *defined* predicates. A predicate is defined if it has *definition clauses*; otherwise it is free. A definition clause has the form ' $H :- B$ '. H is an atomic formula called the *head* of this definition clause, and B is a conjunction of literals. A definition clause of a defined predicate is one whose head has that predicate. The approximate logical meaning² of a defined predicate is given in terms of a necessary and sufficient condition obtained from its definition clauses. For instance, if predicate p has the two definition clauses in (8), its meaning is roughly defined by (9).

$$(8) p(X) :- q(X, a). \quad p(f(X)) :- r(X).$$

$$(9) \forall A \{p(A) \Leftrightarrow \{\exists Y (q(A, Y) \wedge Y = a) \vee \exists X (A = f(X) \wedge r(X))\}\}$$

Predicates like $=f$ as in atomic formula (binding) $X=f(Y)$ are free, and thus variable bindings are typical free atomic formulas. Most features are also free predicates, and thus most feature specifications are free atomic formulas. Computation may stop when the current hypothesis contains only free literals and there is no need for consistency checking.

¹A more serious account of information processing could be based on transformation of the entire constraint instead of simply rewriting the top-level hypothesis. For further details, see Hasida and Ishizaki (1987), Tuda, Hasida and Sirai (1989), and Hasida and Tsuda (1991), among others.

²Potential energy underlies the full meaning of the constraints, which is much more delicate than the logical meaning.

Out of various aspects of potential energy, in this paper we consider only what we will call the *cost* of literals and hypotheses. Like *assumability cost* of Hobbs et al. (1990), our cost is the cost of maintaining hypotheses. That is, a cost increases potential energy and thus degrades the plausibility of the relevant hypothesis. Smaller cost corresponds to better interpretation of a sentence, for example. Thus a hypothesis with a large cost should be transformed through inferences so as to reduce the cost, or the hypothesis will be abandoned. For simplicity, we assume that costs are primarily assigned to literals in the top-level hypothesis, and that the cost of the entire hypothesis is the sum of the costs of the literals it contains. In this section we do not care about small differences of the cost, but just distinguish very costly literals by emboldening their predicates, like $\mathbf{p}(\mathbf{X})$.

Throughout the rest of the paper, we employ a very tentative and simplified version of cost dynamics. That is, when two literals with costs α and β unify (factor), the resulting literal has cost $\min(\alpha, \beta)$. This means that literals with larger costs should unify with others with smaller costs, so that the overall cost of the hypothesis should be reduced.

Note that costs cannot be eliminated freely, because the computational scheme described above does not allow free introduction of assumptions whether or not those assumptions are known to be consistent with the current hypothesis. Introduction of any literal must be based explicitly on some clause. Thus, attempt of cost elimination triggers nontrivial inferences. At any rate, it is in order to capture the behavior of costs that we should be so explicit about representation and computation as we are above.

3 Encoding Linguistic Constraints

Before entering the main part of the paper, some introductory comments are in order about how to represent constraints pertaining to natural language.

The domain of interpretation of our first order language includes the domain of syntactic objects and the domain of semantic objects of natural language. Let us refer to the former as SYN and the latter as SEM.

We organize SYN incorporating methods of the constraint-based grammars concerning the use of record-like structures. SYN includes grammatical categories and other syntactic constructs. Grammatical categories are represented in terms of records (alias feature bundles or attribute-value pairs). For instance, the following grammatical category says that the part of speech of the linguistic (Japanese, in this case) expression in question is postposition of form 'ga,' and it is directly associated with a semantic object X.

(10) {pos/p,form/ga,sem/X}

pos, form, and sem are features. p and ga are syntactic objects that are not categories.

SEM is organized on the basis of Situation Theory and Situation Semantics (Barwise and Perry 1983, Barwise and Etchemendy 1987, Barwise 1989), with some amalgamation with Discourse Representation Theory (DRT hereafter; Kamp 1981, 1988). SEM includes *situations*, *states of affairs* (soas henceforth), *types*, *individuals*, and so on. *Collections* include types, quantified objects and perhaps situations, and regarded as sets, masses, and other sorts of complex objects. Some semantic objects are members or parts of collections. Atomic formula $\alpha \in \beta$ means that semantic object α is a member of another semantic object β , and $\alpha \sqsubset \beta$ means that α is a subset or a part of β . Binary relation \sim is the union of \in and \sqsubset . That is, $\alpha \sim \beta$ holds iff either $\alpha \in \beta$ or $\alpha \sqsubset \beta$ holds. Note that α

and β here are first order individuals, and that the infix operators \in , \sqsubset , and \rightsquigarrow are first order predicates.

Soas are record-like semantic objects of the form $\{\rightsquigarrow/\alpha\}$, where α is a type. For instance, soa $\langle\langle\text{kiss, kisser:john, kissed:mary}\rangle\rangle$ and $\langle\langle\text{drink, drinker:X, drunk:Y}\rangle\rangle$ are regarded as (11) and (12), respectively.

(11) $\{\in/\text{kiss,kisser/john,kissed/mary}\}$

(12) $\{\sqsubset/\text{drink,drinker/X,drunk/Y}\}$

A kissing event, which is atomic, is a member of type *kiss*, whereas a drinking event is a part of *drink*. Binary predicates such as *kisser*, *kissed*, *drinker*, and *drunk* are called *roles*. We remain undecided about whether roles are features.³ We do not assume that soas carry polarity of Situation Theory.⁴ Unlike in the other versions of Situation Theory, a dog is represented as a soa $\langle\langle\text{dog}\rangle\rangle$: that is, an individual α such that $\alpha \in \text{dog}$. Some quantity of water is also a soa $\langle\langle\text{water}\rangle\rangle$, but it is $\{\sqsubset/\text{water}\}$ rather than $\{\in/\text{water}\}$.

Incorporating an aspect of DRT, we postulate that there is a quasi-order relation⁵ \preceq in SEM which may be regarded as the reflexive and transitive closure of the scoping relation in the traditional sense. So $\alpha \preceq \beta$ means that α belongs to a scope smaller than or equal to the one β belongs to. The definition of \preceq and related constraints are listed below.

(13) $X \preceq Y :- X \simeq Y.$

(14) $X \preceq Y :- X \prec Y.$

(15) $X \prec W :- X \simeq Y \wedge \text{col}(Y,Z) \wedge Z \preceq W.$

(16) $X \prec_1 W :- X \simeq Y \wedge \text{col}(Y,Z) \wedge Z \simeq W.$

(17) $\{\rho/\text{ARG}\} \preceq \text{ARG}.$ (for each role ρ)

(18) $X \rightsquigarrow Y \leftarrow \text{col}(X,Y).$

(13) and (14) together are to the effect that \preceq is the transitive closure of $\simeq \cup \text{col}$. \simeq is an equivalence relation in SEM. $\alpha \simeq \beta$ means that α and β belong to the same minimal scope of quantification (or abstraction). That is, each quantifier scope may be looked upon as an equivalence of \simeq . $\text{col}(\alpha,\beta)$ means that α is the representative element or part of collection β . col is a special feature whose inverse is also a partial function. That is, if $\text{col}(\alpha,\beta) \wedge \text{col}(\gamma,\delta)$ then either $\alpha=\gamma \wedge \beta=\delta$ or $\alpha \neq \gamma \wedge \beta \neq \delta$. So if a collection has a scope, it has just one representative element or part in the scope. Each semantic object is the representative of at most one collection. (17) is the accessibility constraint. Both should be expected similarly because equivalence classes of \simeq are scopes of quantification or abstraction in general.

(19) and (20) jointly mean that SEM forms a tree (or trees) branching only downwards, as should be expected from the viewpoint that \preceq is the scoping relation.

³If they are, there is exactly one lover for each loving event, for instance.

⁴Although we do not spell out further details, we can represent negation more along traditional lines, assigning a scope to each negation. This still leaves at least three different affirmation statuses for a soa: true, don't care, and false. The variety turns out larger when we take cost assignments into account.

⁵A binary relation R is a quasi-order iff it is both reflexive (xRx for every x) and transitive (if xRy and yRz then xRz).

$$(19) \neg X \simeq Y \vee \neg X \prec Y.$$

$$(20) X \simeq Y \leftarrow U \simeq V \wedge \text{col}(U, X) \wedge \text{col}(V, Y).$$

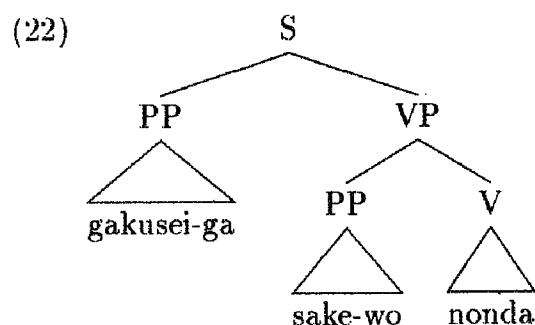
Each node of this tree is an equivalence class of \simeq . Node A is the mother node of B iff there are $\alpha \beta \in \text{SEM}$ such that $\alpha \in A$, $\beta \in B$, and $\text{col}(\alpha, \beta)$. Due to (19), we have $\prec = \preceq - \simeq$ (there may be more than one combination of such α and β). So $\alpha \prec_1 \beta$ means that α and β belong to two adjacent equivalence classes of \simeq . (19) implies that col must always encompass two distinct equivalence classes of \simeq , so that \prec constitutes no cycle,⁶ (20) means that \preceq does not fork upwards into two distinct equivalence classes of \simeq .

4 Local Dependency

Here we take a look at a role our cost may play in putting smaller expressions together to form bigger ones. Consider the following Japanese sentence.⁷

- (21) Gakusei -ga sake -wo nonda.
 student NOM sake ACC drank
 'A student drank sake.'

In constituency-based approaches, including GPSG, HPSG, JPSG, LFG, and transformational grammars, this sentence will be assigned a syntactic structure in (22). Given this, a

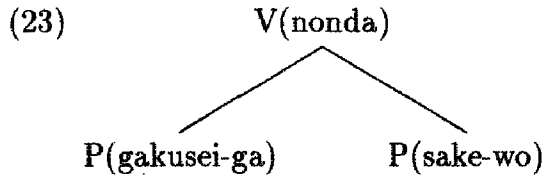


suspicion might arise: perhaps the grammatical categories such as VP and S here encode intermediate steps in the computation to work out the entire sentential structure. That is, the relationship between a category and its mother in a local tree might be similar to the relationship between the input and the output structural descriptions in transformation. If this suspicion turns out true, and if this case is parallel to the case with transformations, then it should be possible to recast the relationship between a category and its mother to a relationship between two different states of computation probably just before and after an operation to set up local dependency, so that the head-mother distinction be abandoned. Such an approach has been called dependency grammars (Sgall and Panevová 1989). In this viewpoint, sentence (21) has the structure in (23).⁸ An edge in this dependency tree

⁶The circularity as in the Liar sentence (Barwise and Etchemendy 1987) can obtain within an equivalence class of \simeq .

⁷The English translation of *gakusei* may also be 'the student,' 'students' or 'the students'; the Japanese language does not usually mark definiteness and number.

⁸Throughout the paper, we treat each postpositional phrase as a word for simplicity.



means that the category at its lower end depends on the category at the upper end. The comparison of the two structures in terms of the number of the categories suggests that the dependency-based approach will reduce the combinatorial complexity of the theory.

The mother-head relationship in the constituency-based account is captured in dependency-based account by using cost. To show this, let us first postulate the following dependency-based rule for head-final constructions.

- (24) $\text{expr}(\text{Head}, \text{Left}, \text{Right}) :-$
 $\text{expr}(\text{Cat}, \text{Left}, \text{Middle}) \wedge \text{expr}(\text{Head}, \text{Middle}, \text{Right}) \wedge \text{dep}(\text{Cat}, \text{Head}).$

Predicate *expr* means that the first argument is the grammatical category of the linguistic expression beginning at the point designated by the second argument and ending at the point designated by the third argument. Note that this rule does not introduce any mother category as separate from the head category. *dep* is a free predicate, and means that the first argument locally depends on the second argument. We tentatively assume that *dep* is a feature. So every expression depends on at most one other expression.

To account for sentence (21), we postulate the following lexical entries.⁹

- (25) $\text{expr}(\text{Cat}, \text{"gakusei-ga"}) :-$
 $\text{Cat} = \{\text{pos/p, form/ga, sem}/\{\sim/X\}\} \wedge \text{dep}(\text{Cat}, \{\text{pos/v}\}).$
- (26) $\text{expr}(\text{Cat}, \text{"sake-wo"}) :-$
 $\text{Cat} = \{\text{pos/p, form/wo, sem}/\{\sim/X\}\} \wedge \text{dep}(\text{Cat}, \{\text{pos/v}\}).$
- (27) $\text{expr}(\text{Cat}, \text{"nonda"}) :-$
 $\text{Cat} = \{\text{pos/v, sem/DRANK}\} \wedge$
 $\text{col}^*(\langle\langle\text{drank, drinker:X, drunk:Y}\rangle\rangle, \text{DRANK}) \wedge$
 $\text{dep}(\{\text{pos/p, form/ga, sem/X}\}, \text{Cat}) \wedge$
 $\text{dep}(\{\text{pos/p, form/wo, sem/Y}\}, \text{Cat}).$

The apparently second and last argument of *expr*(...) is the part of the utterance between the two points designated by the actual second and the third arguments. *col** is the reflexive and transitive closure of *col*. As discussed later, the literal of the form *col**(...) in (27) makes room for optional quantifications of arbitrary depth. The intuitive meaning of the cost assignment here is: *gakusei-ga* and *sake-wo* should depend on some verbal category, and *nonda* should be modified by a *ga*-phrase and a *wo*-phrase.

The interpretation of sentence (21) begins with the following unit hypothesis, which means that *gakusei-ga sake-wo nonda* should be a well-formed linguistic expression.

- (28) $\text{expr}(\text{Cat}, \text{"gakusei-ga sake-wo nonda"})$

⁹Note the form of the semantic object associated with *gakusei-ga* (essentially *gakusei*). This reflects the fact that the Japanese language lacks morphological marking of singularity, as mentioned in Note 7.

We consider that to interpret (21) is to derive from (28) a new hypothesis (approximately) all of whose implications are consistent with the given constraints. Let us call such a new hypothesis an *interpretation* of (21). To get an interpretation, (28) is transformed into the following through resolution with (25), (26) and (27).¹⁰

- (29) $\text{Cat0} = \{\text{pos/p, form/ga, sem}/\{\sim/\text{student}\}\} \wedge \text{dep}(\text{Cat0}, \text{Cat}) \wedge$
 $\text{Cat1} = \{\text{pos/p, form/wo, sem}/\{\sim/\text{sake}\}\} \wedge \text{dep}(\text{Cat1}, \text{Cat}) \wedge$
 $\text{Cat} = \{\text{pos/v, sem/DRANK}\} \wedge \text{col}^*(\langle\langle\text{drank, drinker:X, drunk:Y}\rangle\rangle, \text{DRANK}) \wedge$
 $\text{dep}(\{\text{pos/p, form/ga, sem/X}\}, \text{Cat}) \wedge$
 $\text{dep}(\{\text{pos/p, form/wo, sem/Y}\}, \text{Cat}).$

The compulsory unifications among literals of the form $\text{dep}(\dots)$ have been finished here, leaving two literals $\text{dep}(\text{Cat0}, \text{Cat})$ and $\text{dep}(\text{Cat1}, \text{Cat})$. A better interpretation is one with smaller cost. The minimum cost is obtained by further factoring the literals of the form $\text{dep}(\dots)$ in the expected combination, and accordingly unifying other literals.¹¹

- (30) $\text{dep}(\{\text{pos/p, form/ga, sem/X}\}, \text{Cat}) \wedge X \sim \text{student} \wedge$
 $\text{dep}(\{\text{pos/p, form/wo, sem/Y}\}, \text{Cat}) \wedge Y \sim \text{sake} \wedge$
 $\text{Cat} = \{\text{pos/v, sem/DRANK}\} \wedge \text{col}^*(\langle\langle\text{drank, drinker:X, drunk:Y}\rangle\rangle, \text{DRANK}).$

This interpretation subsumes various quantificational possibilities, to which we return in the next section.

As is illustrated by this example, the above grammar fragment captures the following principle.

- (31) A specified argument place of a verb requires a complement to be associated with it.¹²

This principle is embodied in terms of the literals of the form $\text{dep}(\alpha, \text{Cat})$ in (27) and (29). These literals are assigned costs, so that they must unify with other literals. It is easy to see that the expected combination of unifications is the only possible one in the case of (29). Even in more complicated cases where there are other literals of the form $\text{dep}(\alpha, \beta)$ with β distinct from Cat , the expected unifications are as strongly preferred, provided another principle as follows.

- (32) Two grammatical categories do not unify when they are associated with different parts of utterance.

This is regarded as a very general principle. In fact, it also predicts the badness of the following.

- (33) Gakusei -ga gakusei -ga kita.
 student NOM student NOM came
 'A student a student came.'

¹⁰We have omitted the resolved literals, though they might be necessary for consistency checking.

¹¹We have done resolution entirely before cost-driven factoring here, but this processing order is just for explanatory ease. Of course it is possible to interpret the sentence on a more incremental basis, for instance.

¹²The optionality of complements in Japanese may be formulated in terms of a lexical rule which deletes argument slots from a verb, for instance.

A straightforward way to implement principle (32) is to assign distinct indices to categories to be distinguished in terms of feature specifications for feature, say, *ind*. Indices can simply be associated with positions in the utterance.

Note also that the above grammar fragment captures the freedom of the order among the elements (complements and adjuncts) locally depending on a verb. For instance, it allows *sake-wo gakusei-ga nonda* as well as *gakusei-ga sake-wo nonda*. In this connection, constraints on word order, if any, may be stated separately.

Owing to the cost-based component, our account of complementation is much simpler than constituency-based accounts, with respect to combinatorial complexity. For instance, GPSG and HPSG implement (31) and (32) in terms of **subcat** feature and the Subcat Feature Principle (SFP), both of which are of combinatorial nature. The value of **subcat** feature is a list, a combinatorial syntactic object, and the application of SFP accompanies combinatorial operations where a new list is made by deleting (or inserting, if you look at it on a top-down basis) an element in a **subcat** list, in order to embody (32). (31) is captured by requiring that **subcat** lists at the maximally projected categories should be empty except for the controlled elements. JPSG assumes that the **subcat** value is a set instead of a list. Although a set might appear less combinatorially complex than a list, the application of SFP is no less complicated with a set than with a list; you must look at possibly all the elements of a set when you tailor another set from it by adding or subtracting some elements. In this respect, our account presented above involves unifications among categories but no unifications among lists or sets.

Further, our approach naturally extends to adjunction as well. So we have adopted the general name for predicate **dep**, in a sense neutral between complementation and adjunction. In the case of adjunction, the lexical entry of the verb does not contribute a literal of the form **dep**(...), so that the thematic role of the adjunct is determined by the adjunct itself and the semantic property of the verb. Thus complementation and adjunction are given very analogous accounts. Namely, both phenomena are accounted for equally in terms of unifications among literals of the form **dep**(...), and the only difference between them is that the verb contributes some syntactic information to determine the thematic role of a complement but not that of an adjunct. So our approach contrasts again with the accounts based on **subcat** feature, which sharply distinguish between complements and adjuncts, in the sense that complementation is accounted for in terms of **subcat** feature and adjunction in terms of some other feature.

Incidentally, such a treatment of local dependency fits the intuition that complementation is less costly (in a pretheoretic sense) and hence more preferable than adjunction, because the former reduces cost but the latter does not.¹³ Some delicate linguistic phenomena are accounted for on the basis of the different processing loads with complementation and adjunction (Hasida 1988, among others). Our current framework might provide a formal basis for these accounts.

So called θ -criterion is formulated in terms of cost assignment, as a purely semantic rather than syntactic constraint. For example, the semantic structure of *wo*-phrase of Japanese (postpositional phrase having the accusative case by default) might include a literal of the form **patient**(α, β). β is the **sem** value of this *wo*-phrase. **patient** is a general role subsuming *loved*, *drunk*, and so on. That is, we have constraints like the following.

(34) **patient**(X,Y) \leftarrow **loved**(X,Y).

¹³The author is grateful to the anonymous referee for drawing attention to this respect.

All this means that *wo*-phrase requires a soa to which it supplies a patient-type role parameter.¹⁴ In a typical case of the computation triggered by the above cost, a soa with a patient-type role such as *loved* is found, a rule such as (34) is applied to yield a literal of the form *patient*(...) without cost, and this literal is unified with *patient*(...) to eliminate the cost. In this connection, our approach will also contribute to the investigation of children's language use and more drastic attempts to minimize syntax (Suzuki and Tutiya 1989), in which syntactic constraints do not play major roles.

5 Quantifier Scoping

The discussion in the previous section has been motivated by the suspicion that constituent structures encode intermediate states of computation which should be needless in a genuine constraint-based framework. A similar and probably stronger suspicion arises regarding a prevailing treatment of quantifier scoping. That is, quantifier storage (Cooper 1983, Keller 1988) appears to explicitly encode intermediate states of computation (Hobbs and Shieber 1987), and is very probably dispensable. Here we propose a simple treatment of quantifier scoping without any explicit storage mechanism. In doing so, we pay attention to the quasi-order relation \preceq , which corresponds to the scoping relation, as mentioned already.

Japanese sentence (35)¹⁵ has two plausible (minimal) readings, (36) and (37), of which the former is the default interpretation.

- (35) Gakusei-ga san -nin sake-wo ro -ppon nonda.
 student NOM three person sake ACC six bottle drank
 '3 students drank 6 bottles of sake.'
- (36) 3 students and 6 bottles of sake were involved in a drinking event. (cumulative reading)
- (37) Each of 3 students drank 6 bottles of sake. (wide scope reading of *san-nin*)

To account for this, we further postulate the following lexical entries for the floating quantifiers *san-nin* and *ro-ppon*, plus some additional constraints.

- (38) $\text{expr}(\text{QCat}, "san-nin") :-$
 $\text{QCat} = \{\text{pos}/n, \text{sem}/\text{QSem}\} \wedge \text{dep}(\text{QCat}, \text{VCat}) \wedge$
 $\text{PCat} = \{\text{pos}/p, \text{form}/F, \text{sem}/\text{PSem}\} \wedge \text{ga-wo}(F) \wedge \text{dep}(\text{PCat}, \text{VCat}) \wedge$
 $\text{VCat} = \{\text{pos}/v, \text{sem}/\text{VSem}\} \wedge$
 $\text{quant}(\text{PSem}, 3\text{nin}) \wedge \text{PSem} \prec_1 \text{QSem} \wedge \text{col}^*(\text{QSem}, \text{VSem}).$
- (39) $\text{expr}(\text{QCat}, "ro-ppon") :-$
 $\text{QCat} = \{\text{pos}/n, \text{sem}/\text{QSem}\} \wedge \text{dep}(\text{QCat}, \text{VCat}) \wedge$
 $\text{PCat} = \{\text{pos}/p, \text{form}/F, \text{sem}/\text{PSem}\} \wedge \text{ga-wo}(F) \wedge \text{dep}(\text{PCat}, \text{VCat}) \wedge$
 $\text{VCat} = \{\text{pos}/v, \text{sem}/\text{VSem}\} \wedge$
 $\text{quant}(\text{PSem}, 6\text{pon}) \wedge \text{PSem} \prec_1 \text{QSem} \wedge \text{col}^*(\text{QSem}, \text{VSem}).$

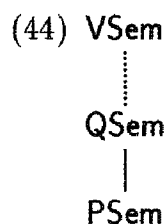
¹⁴Of course the story would not be that simple if we take into consideration *wo*-phrases which do not play a patient role, as in *sora-wo tobu* 'fly in the sky.'

¹⁵*-nin* and *-hon* are classifiers of Japanese, for counting people and long objects, respectively. The latter becomes *-ppon* after some numerals.

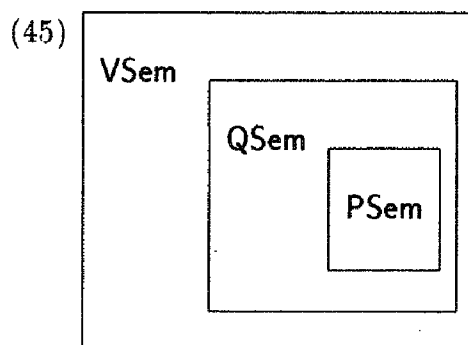
- (40) $ga_wo(F) :- F=ga.$
 (41) $ga_wo(F) :- F=wo.$
 (42) $col^*(X,Y) :- X=Y.$
 (43) $col^*(X,Z) :- col(X,Y) \wedge col^*(Y,Z).$

Here a floating quantifier is regarded as an adjunct of a verb. Due to the general dependency rule (24), $VCat$ is to unify with the category of the verb. Just as with complementation, literal $dep(PCat,VCat)$ is to unify with another literal with dep to eliminate the cost. As a result, $PCat$ is to unify with the category of a *ga*-phrase or *wo*-phrase which depends on the verb.

The semantic structure encoded in the body of (38) and (39) may be depicted as in (44), if we pay attention to scoping relationships only.



Relation \preceq is represented by dotted lines, and \prec_1 by solid perpendicular lines. The lower end of each such line is smaller than (\prec) or equivalent to (\simeq) the higher end. The dotted line in (44) reflects $QSEM \preceq VSEM$, which is implied by $col^*(QSEM,VSEM)$. In the box notation of DRT, (44) would look like the following.



Each box corresponds to an equivalence class of \simeq . The outermost box and the second may be identical, $VSem$ and $QSem$ belonging to the same minimal box; otherwise there may be zero or more boxes between the two outer boxes.

$quant(\alpha,\beta)$ means that α may be instantiated so much as to reach quantity β . In the case of (38), for instance, copy of *Student* may produce up to three different instances of *Student*. β may also be *every*, *most*, and so on, but further details are irrelevant to the current purpose. $QSem$ stands for the minimal quantified event involving three people in the case of (38) and six bottles (or long objects in general) in the case of (39). $PSem$ is the bound parameter of this quantification, as indicated by $PSem \prec_1 QSem$, and an instantiation of $PSem$ due to $quant(\alpha,\beta)$ accompanies a copy of the whole equivalence class of \simeq containing $PSem$ into the equivalence class containing $QSem$. $col^*(QSem,VSem)$ is a constraint to the effect that a floating quantifier cannot outscope the minimal sentence containing it. Such a constraint may be associated with other syntactic constructs as well.

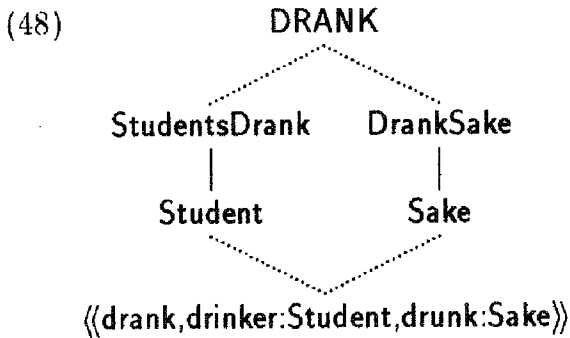
A literal of the form $\text{col}^*(\dots)$ in the lexical entry of a verb (such as (27)) accounts for why a sentence such as *tori-wa tobu* 'a bird flies' may be interpreted as if it involved a universal quantifier. Although (43) allows infinitely many different interpretations, it is not the case that just anything goes. $\text{col}^*(\dots)$ in (43) indicates that there should be an abstraction (quantified soas and some types) introduced elsewhere in order to use (43). As seen below, this explains why (35) has the interpretation preference mentioned above.

From (46) we can infer (47) as an abductive explanation, after reducing the cost of the literals of the form $\text{dep}(\dots)$ by factoring.

(46) $\text{expr}(\text{Cat}, \text{"gakusei-ga san-nin sake-wo ro-ppon nonda"})$.

(47) $\text{col}^*(\text{StudentsDrank}, \text{DRANK}) \wedge$
 $\text{col}^*(\text{DrankSake}, \text{DRANK}) \wedge$
 $\text{Student} \prec_1 \text{StudentsDrank} \wedge$
 $\text{Sake} \prec_1 \text{DrankSake} \wedge$
 $\text{Drank} \preceq \text{Student} \wedge$
 $\text{Drank} \preceq \text{Sake} \wedge$
 $\text{col}^*(\text{Drank}, \text{DRANK}) \wedge$
 $\text{Drank} = \langle\langle \text{drank}, \text{drinker:Student}, \text{drunk:Sake} \rangle\rangle \wedge$
 $\text{dep}(\{\text{pos/n}, \text{sem/StudentsDrank}\}, \text{Vcat}) \wedge$
 $\text{dep}(\{\text{pos/n}, \text{sem/DrankSake}\}, \text{Vcat}) \wedge$
 $\text{dep}(\{\text{pos/p}, \text{form/ga}, \text{sem/Student}\}, \text{VCat}) \wedge$
 $\text{dep}(\{\text{pos/p}, \text{form/wo}, \text{sem/Sake}\}, \text{VCat}) \wedge$
 $\text{VCat} = \{\text{pos/v}, \text{sem/DRANK}\}.$

This inference involves deductions by (17), which gave rise to the two literals $\text{Drank} \preceq \text{Student}$ and $\text{Drank} \preceq \text{Sake}$. The constraints on quantifier scoping contained in (47) is pictorialized as follows.



DRANK is the quantified soa associated with the entire sentence, and stands for a complex drinking event involving three students and six bottles of sake. **StudentsDrank** is the quantified soa standing for the minimal drinking event involving three students, and **Student** is the parameter bound by this quantification. Similarly, **DrankSake** is the minimal drinking event involving six bottles of sake, and parameter **Sake** is bound in it. Note that this constraint does not have any concise box representation like (45).

As mentioned earlier, (20) rules out upward forking of \preceq . So all the six nodes in (48) must be linearized in terms of \preceq . This linearization is obtained by further resolutions on the literals with predicates \preceq and col^* together with some factorings and deductions. Let us consider only those interpretations minimal in terms of the number of times clauses

are exploited. Such interpretations come out quickly, gaining preferences. In addition, stronger preferences result from fewer exploitations of clauses introducing costs. So we are interested in minimizing the use of (43) in particular here. That is, we will substitute as many occurrences of *col** with equality as possible.

There are three minimal interpretations worked out accordingly, as shown below with illustrations of the associated scoping relations.

- (49) 3 students and 6 bottles of sake were involved in a drinking event. (cumulative reading)

$$\begin{array}{c} \text{DRANK} = \text{StudentsDrank} = \text{DrankSake} \\ | \\ \langle\langle \text{drank}, \text{drinker:Student}, \text{drunk:Sake} \rangle\rangle \simeq \text{Student} \simeq \text{Sake} \end{array}$$

- (50) Each of 3 students drank 6 bottles of sake. (wide scope reading of *san-nin*)

$$\begin{array}{c} \text{DRANK} = \text{StudentsDrank} \\ | \\ \text{DrankSake} \simeq \text{Student} \\ | \\ \langle\langle \text{drank}, \text{drinker:Student}, \text{drunk:Sake} \rangle\rangle \simeq \text{Sake} \end{array}$$

- (51) Each of 6 bottles of sake was drunk by 3 students. (wide scope reading of *ro-ppon*)

$$\begin{array}{c} \text{DRANK} = \text{DrankSake} \\ | \\ \text{StudentsDrank} \simeq \text{Sake} \\ | \\ \langle\langle \text{drank}, \text{drinker:Student}, \text{drunk:Sake} \rangle\rangle \simeq \text{Student} \end{array}$$

Out of these, only (51) is barred. This seems to be a sort of subject-object asymmetry, but we do not go into details here because it is irrelevant to the purpose of the present paper.

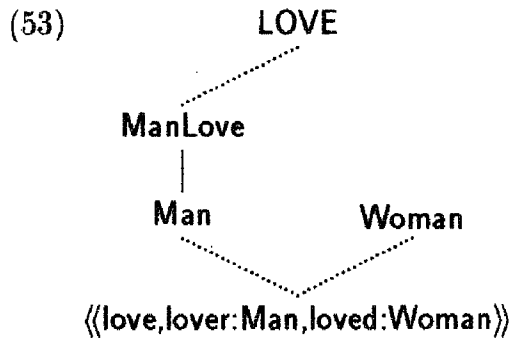
As for the remaining two interpretations, it is natural to consider that (49) (= (36)) defaults to have stronger preference than (50) (= (37)), because the former is a smaller explanation in terms of the number of exploitations of (43). This accounts for why it is the default interpretation. The cost-based dynamic account thus straightforwardly captures some aspects, such as the complexity of inferences, which have been abstracted away from the underlying symbolic logic.

In the above example, the two quantified noun phrases are not related directly and thus all the permutations involving unification are possible, except that just one interpretation is ruled out by subject-object asymmetry. So let us consider some different patterns.

To begin with, we can account for the two possible (minimal) scopings of the following sentence.

- (52) Every man loves a woman.

As mentioned before, the word-order constraints of English may be stated in addition to the local dependency constraints. A universal quantification is assigned a semantic structure totally parallel to the one in (38) and (39). A singular indefinite does not create any quantification but just introduces a parameter, following the DRT approach. Further details are omitted. The scoping relationships involved in a half-cooked interpretation of (52) look like (53). Just similarly to the previous case, LOVE is the semantic object

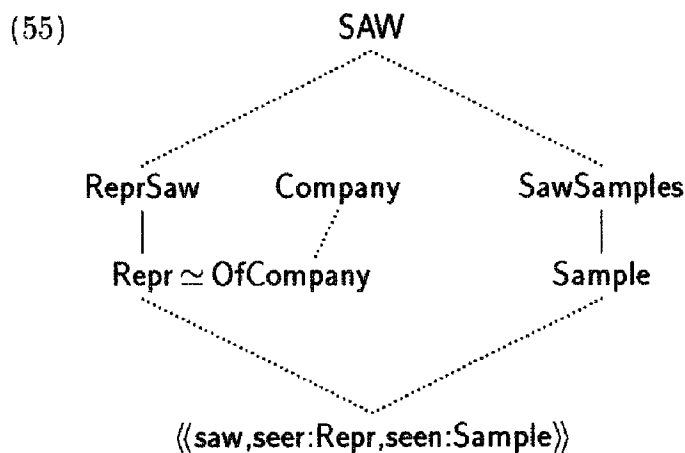


associated with the whole sentence. **ManLove** is the soa quantified by *every*, and **Man** is the parameter bound by this quantifier. The scope to this quantification is the equivalence class of \simeq containing **Man**. **Woman** is the semantic object associated to *a woman*. The linearization of the diagram in (53) forces us to choose between the wide-scope reading and narrow-scope reading of *every*. The former is characterized by $\text{Woman} \leq \text{Man}$, and the latter $\text{ManLove} \leq \text{Woman}$. All the possible cases fall in either of these two.

Next example (taken from Hobbs and Shieber (1987)) is a little more complicated:

(54) Every representative of a company saw most samples.

This sentence cannot be interpreted so that *every representative* outscopes *most samples* and *most samples* outscopes *a company*. So there are just five rather than six (=3!) different minimal scopings here. This is accounted for as follows in the present approach. The constraints on scopings look like (55). Although we do not go into details here,



OfCompany, Company, $\text{Repr} \simeq \text{OfCompany}$ are introduced by copying from the content

of the type associated with *representative of a company*; the same mechanism underlying so called E-type anaphora. *OfCompany* is a copy of the soa associated with *of a company*, and *Company* is the corresponding copy of the parameter associated with *a company*. *OfCompany* will look like $\langle\langle\text{represent, representative:Repr, represented:Company}\rangle\rangle$. $\text{Repr} \simeq \text{OfCompany}$ holds because *of a company* is a restrictive modification to *representative*. This characterization of restriction is essentially the same as *Absorption Principle* of Gawron and Peters (1990).

Again (55) must be linearized. But note that *SawSamples* cannot cut in between *ReprSaw* and *Company* in the resulting linear ordering, because either $\text{Company} \simeq \text{Repr}$ or $\text{ReprSaw} \preceq \text{Company}$ holds.

After all, our account of quantification does not involve quantifier storage. The quasi-order relation encoding constraints on quantifier scoping is directly handled instead. In this connection, note that our approach exploits minimal syntactic structures. As we have discussed, we postulate a dependency hierarchy but no constituency hierarchy of syntax. This is all right because we have enough structure in semantics, such as the quasi-order hierarchy of semantic objects.

6 Concluding Remarks

A grammar fragment has been worked out to demonstrate that combinatorial complexity of constraint-based linguistic theories can be reduced by eliminating explicit encoding of intermediate states of computation. This is along the same line on which transformational rules have been abandoned in those theories. We have discussed how to eliminate *subcat* feature, constituent structure, and quantifier storage. Subjects of further investigation will include binding features such as *slash* and *que*. The chances are that they will be reduced to a large extent, if not entirely eliminated.

We have employed a notion of cost, mainly in dealing with local dependencies and, in a somewhat complementary manner, in handling quantifier scopings. Cost is a part of potential energy, which constitutes the analog aspect of constraint. Although here we have used this analog aspect just to obtain truth values (alias preferences of interpretation), potential energy is quite naturally used for providing preferences of computation, based on energy minimization principle.

Such a unified treatment of truth conditions and computation control is not simply due to the convenience in theory construction, but necessitated by more essential reasons. That is, since the standard methods of defining truth conditions (or consistency) are computationally intractable, truth values can be obtained at best as preferences of interpretation depending on local computational contexts. Under this intractability, purely logical approaches to truth conditions, typically represented by non-monotonic logics (McCarthy 1980, McDermott and Doyle 1980), are useless as long as they disregard partial processing.

From a viewpoint of linguistics, this direction is to eliminate the distinction of linguistic competence (concerning determination of truth values) and linguistic performance (concerning control of computation). From a wider perspective, we face a need for restructure descriptive formalisms, such as Situation Theory, so that the notion of truth and consistency should be defined in tight conjunction with processing control. It is a small first step into this direction that we have marked in this paper.

References

- Barwise, J., and Perry, J. 1983. *Situations and Attitudes*, Cambridge: MIT Press.
- Barwise, J., and Etchemendy, J. 1987. *The Liar: An Essay on Truth and Circular Propositions*, New York: Oxford University Press.
- Barwise, J. 1989. *The Situation in Logic*, CSLI Lecture Notes Number 17, Stanford: CSLI Publications.
- Bresnan, J. ed. 1982. *The Mental Representation of Grammatical Relations*, Cambridge: MIT Press.
- Chomsky, N. 1981. *Lectures on Government and Binding*, Dordrecht: Foris.
- Chomsky, N. 1986. *Barriers*, Linguistic Inquiry Monograph 13, Cambridge: MIT Press.
- Cooper, R. 1983. *Quantification and Syntactic Theory*, Synthese Language Library Volume 21, Dordrecht: Reidel.
- Gazdar, G., Klein, E., Pullum, G. K., and Sag, I. A. 1985. *Generalized Phrase Structure Grammar* Cambridge: Harvard University Press.
- Gawron, J. M., and Peters, S. 1990. *Anaphora and Quantification in Situation Semantics*, CSLI Lecture Notes Number 19. Stanford: CSLI Publications.
- Gunji, T. 1987. *Japanese Phrase Structure Grammar*, Dordrecht: Reidel.
- Hasida, K., and Ishizaki, S. 1987. Dependency Propagation: A Unified Theory of Sentence Comprehension and Generation, *Proceedings of the 10th International Joint Conference on Artificial Intelligence*, 664-670.
- Hasida, K. 1988. A Cognitive Account of Unbounded Dependency, *Proceedings of the 12th International Conference on Computational Linguistics*, 231-236, Budapest.
- Hasida, K., and Tuda, H. 1991. Parsing without Parser, *Proceedings of the 2nd International Workshop on Parsing Technologies*, 1-10, Cancun.
- Hobbs, J. R., and Shieber, S. M. 1987. An Algorithm for Generating Quantifier Scopings, *Computational Linguistics*, 13: 47-53.
- Hobbs, J. R., Stickel, M., Appelt, D., and Martin, P. 1990. *Interpretation as Abduction*, Technical Note 499, Menlo Park: SRI International.
- Johnson, M. 1988. *Attribute-Value Logic and the Theory of Grammar*, CSLI Lecture Notes Number 16, Stanford: CSLI Publications.
- Kamp, H. 1981. A Theory of Truth and Semantic Representation, In *Formal Methods in the Study of Language*, ed. Groenendijk, J., Janssen, T., and Stokhof, M. Amsterdam: Mathematical Centre.
- Kamp, H. 1988. Discourse Representation Theory: What it is and Where it Ought to Go, In *Natural Language at the Computer (Proceedings of Scientific Symposium of Syntax and Semantics for Text Processing and Man-Machine Communication)*, Blaser, A. ed., Lecture Notes in Computer Science 320, Springer.
- Keller, W. R. 1988. Nested Cooper Storage: The Proper Treatment of Quantification in Ordinary Noun Phrases. In *Natural Language and Linguistic Theories*, Reyle, U., and Rohrer, C. ed. 432-447, Dordrecht: Reidel.

- McCarthy, J. 1980. Circumscription: A Form of Non-Monotonic Reasoning, *Artificial Intelligence*, 13:27-39.
- McDermott, D., and Doyle, J. 1980. Non-Monotonic Logic I, *Artificial Intelligence*, 13:41-72.
- Pollard, C., and Sag, I. A. 1987. *Information-Based Syntax and Semantics, Volume 1 Fundamentals*, CSLI Lecture Notes Number 13. Stanford: CSLI Publications.
- Sgall, P., and Panevová, J. 1989. Dependency Syntax — A Challenge, *Theoretical Linguistics*, 15:73-86.
- Suzuki, H., and Tutiya, S. 1989. A Strictly Incremental Approach of Japanese Grammar, presented at the 1st Conference on Situation Theory and its Applications, Asilomar.
- Tuda, H., Hasida, K., and Sirai, H. 1989. JPSG Parser on Constraint Logic Programming, *Proceedings of the European Chapter of ACL'89*.

Common Heuristics for Parsing, Generation, and Whatever ... *

HASIDA, Kôiti

Institute for New Generation Computer Technology (ICOT)
Mita Kokusai Bldg. 21F, 1-4-28 Mita, Minato-ku, Tokyo 108 JAPAN
Tel: +81-3-3456-3069, E-mail: hasida@icot.or.jp

Abstract

This paper discusses general heuristics to control computation on symbolic constraints represented in terms of first-order logic programs. These heuristics are totally independent of specific domains and tasks. Efficient computation for sentence parsing and generation automatically emerge from these heuristics, capturing the essence of standard parsing procedures and semantic head-driven generation. Thus, the same representation of knowledge, including grammar and lexicon, can be exploited in a multi-directional manner in various aspects of language use.

1 Introduction

One lesson to learn from the repeated failure to design large AI systems in general is that the information flow in the cognitive systems is too complex and diverse to stipulate in the design of these AI systems. To capture this diversity of information flow, therefore, AI systems must be designed at a more abstract level where direction of information flow is not explicit.

This is where *constraint* paradigm comes in. Since constraints do not stipulate the direction of information flow or processing order, constraint-based systems could be tailored to have tractable complexity, unlike procedural systems, which stipulate information flow and thus quickly become too complex for human designers to extend or maintain.

Naturally, the key issue in the constraint-based approach is how to control information flow. A very general control schema independent of any specific domain or task is vitally necessary for the success of this approach.

The present paper introduces a system of constraint in a form of logic program, and a set of very general heuristics to control symbolic operation on the constraints. The symbolic operations here are regarded as transforming logic programs. They are quite permissive operations as a whole, allowing very diverse information processing involving top-down, bottom-up and other directions of information flow. The heuristics control this computation so that only relevant information should be exploited and the resulting

*Presented at the Workshop on Reversible Grammar in Natural Language Processing, held in connection with 29th Annual Meeting of the Association for Computational Linguistics, 17-21 June 1991, University of California, Berkeley. Included in the proceedings, pp.81-90.

representation should be compact. Parsing and generation of sentences are shown to be efficiently done under these heuristics, and a standard parsing algorithm and the semantic head-driven generation [8] naturally emerge thereof.

The rest of the paper is organized as follows. Section 2 describes the syntax of our system of constraint. Section 3 defines the symbolic computation on these constraints, and proposes a set of general heuristics to control computation. Section 4 and Section 5 show how sentence parsing and generation are executed efficiently by those heuristics. Finally, Section 6 concludes the paper.

2 Constraint Network

A program is a set of *clauses*. A clause is a set of *literals*. A *literal* is an *atomic constraint* with a sign in front of it. The sign is a '+', '-', or nil. A literal with a sign '+' or nil is called a *positive literal* and one with a sign '-' is a *negative literal*. An atomic constraint is an *atomic formula* such as $p(X,Y,Z)$, a *binding* such as $X=f(Y)$, a *feature specification* such as $a(X,Y)$, or an *equality* such as $X=Y$. Names beginning with capital letters represent variables, and the other names predicates and functions. A feature specification may be regarded as an atomic formula with a special binary predicate called a *feature*. A feature is a partial function from the first argument to the second argument: that is, if a is a feature and both $a(X,Y)$ and $a(X,Z)$ hold, then $Y=Z$ must also hold. The other atomic constraints may be understood in the standard fashion. The atomic constraints other than equalities are called *proper atomic constraints*.

A clause is written as a sequence of literals it contains followed by a semicolon. The order among literals is not significant. So (1) is a clause, which may also be written as (2).

(1) $-p(U,Y) +q(Z) -U=f(X) -X=Z;$

(2) $+q(Z) -p(f(Z),Y);$

A clause containing a literal with null sign is a *definition clause* of the predicate of that literal. A predicate having definition clauses are called *defined predicate*, and its meaning is defined in terms of completion based on the definition clauses. For instance, if the definition clauses of predicate p are those in (3), the declarative meaning of p is given by (4).

(3) $p(X) -q(X,a); p(f(X)) -r(X);$

(4) $\forall A\{p(A) \Leftrightarrow \{\exists Y(q(A,Y) \wedge Y = a) \vee \exists X(A = f(X) \wedge r(X))\}\}$

A predicate which is not a defined predicate is called a *free predicate*. There is a special 0-ary defined predicate **true**. Its definition clauses are called *top clauses*. A top clause corresponds to the query clause of Prolog, although the latter has **false** instead of **true**.

Programs are regarded as constraint networks. For instance, the following program is a network as in Figure 1.

(i) **true** $-member(a,X);$

(ii) $member(A,[A|S]);$

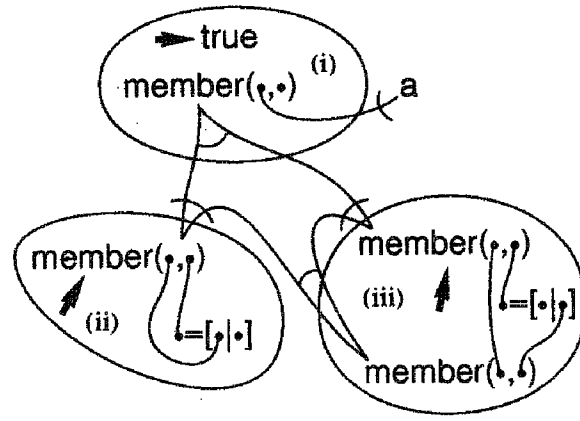


Figure 1: Constraint Network

(iii) $\text{member}(A, [B|S]) \text{ -member}(A, S)$;

In graphical representations like Figure 1, a ‘ \bullet ’ often represents an argument of an atomic constraint. There are two types of *nodes*: arguments, and proper atomic constraints. An argument is involved in at most one proper atomic constraint, but in any number of equalities. An argument bound to a constant is identified with that constant. That is, the first argument of a binding $\bullet = a$, for instance, is represented simply by a . A *link*, represented as a curve, connects two nodes. For any two (possibly the same) nodes, there is at most one link connecting them. A link connecting two arguments is an equality between them. A link connecting two proper atomic constraints is called an *inference link*. No link connects an argument and an atomic constraint. Although often not explicitly shown, an inference link accompanies equalities between the corresponding arguments of the two proper atomic constraints. A *clausal domain* of clause Φ is the part of the constraint network consisting of the atomic constraints referred to as literals in Φ except equalities concerning constants. A clausal domain is depicted by a closed curve enclosing the included atomic constraints. The short thick arrows indicate the references to the atomic constraints as positive literals in clauses. A *predicate domain* of predicate π consists of all the proper atomic constraints with π (binding $X=f(Y)$ is regarded as having binary free predicate $=f$, for instance), inference links among them, and equalities accompanying these inference links.

The *instantiation possibilities* of the constraint network is defined by regarding nodes and links as sets. Those sets are disjoint of each other. An instance of an argument corresponds to an individual in the domain of interpretation, and an instance of an atomic constraint corresponds to an atomic proposition. Constants (bindings to constants) and 0-ary atomic formulas are singleton sets. A link δ between nodes α and β stands for a symmetric relation. That is, $\delta = R \cup R^{-1}$ for some relation $R \subseteq \alpha \times \beta$. We call $\{x \in \alpha | \exists y x \delta y\}$ the α -domain of δ . Every link in a clausal domain or the predicate domain of a defined predicate is of the form $R \cup R^{-1}$ for some bijection R . Let Δ be the transitive closure of the union of all the links. $x \Delta y$ means that x and y correspond to the same object in the domain of interpretation if x and y belong to arguments, and that they correspond to the same atomic proposition if they belong to proper atomic constraints. We say that node α *subsumes* node β when $\alpha / \Delta \supseteq \beta / \Delta$; that is, for every $y \in \beta$ there exists $x \in \alpha$ such that $x \Delta y$. For each pair of a proper atomic constraint α and an argument β of α , there is a bijection ρ from α to β , such that $x \rho y$ holds iff $y \in \beta$ is an argument of $x \in \alpha$.

ρ is called a *role assignment*.

Consider a part \mathcal{P} of the constraint network and the minimum equivalence relation including the links and the role assignments in \mathcal{P} . A *layer* of \mathcal{P} is an equivalence class with respect to this relation. A *splitting domain* is a part \mathcal{S} of the network in which every link is of the form $R \cup R^{-1}$ where R is the union of $(\alpha \cap \mathcal{L}) \times (\beta \cap \mathcal{L})$ over all the layers \mathcal{L} of \mathcal{S} and α and β are the two endnodes of that link. Thus, if a link in a splitting domain splits into two links sharing an endnode α and having disjoint α -domains, then the entire splitting domain splits into two separate splitting domains each containing one of these two links. The clausal domains and predicate domains are assumed to be splitting domains.

A *joint* is a part of a node which connects the node with a link or more. Figure 2 shows

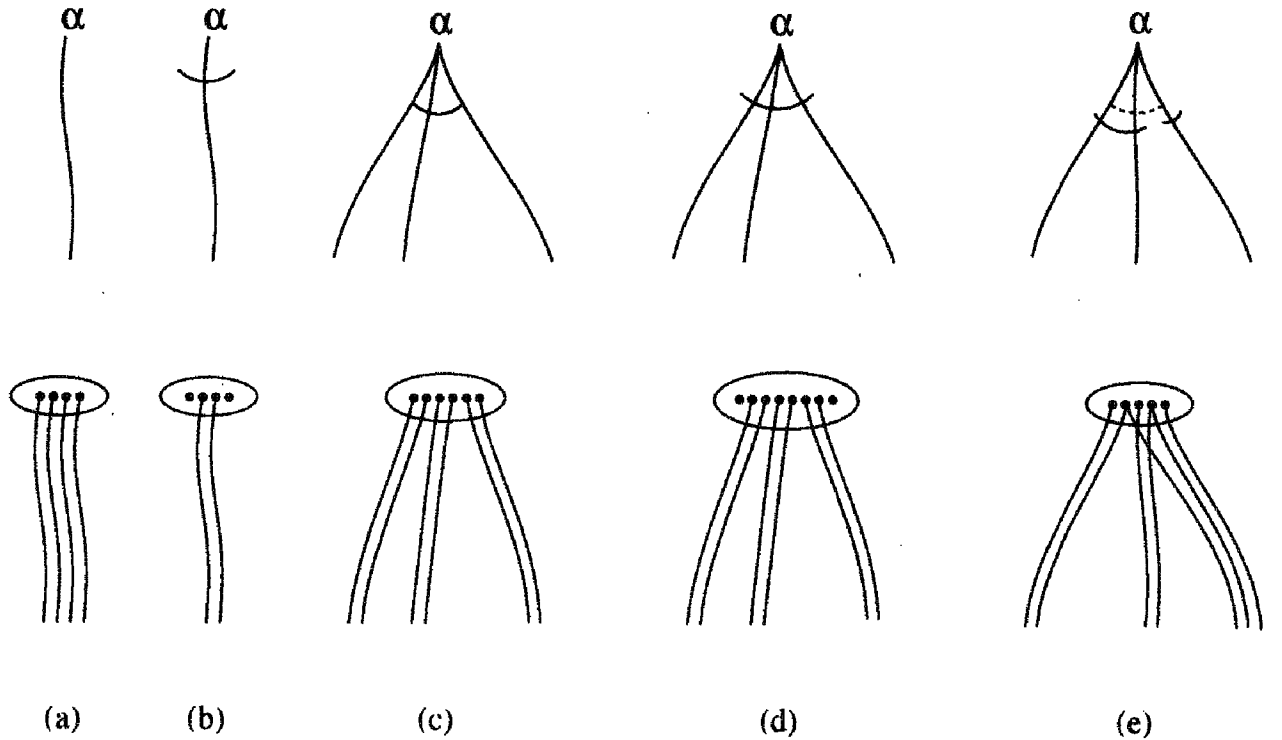


Figure 2: Joints between Nodes and Links

some joints. The figures below illustrate the instantiation possibilities of the networks shown above by depicting each node as an ellipse enclosing its instances, and each link as a bundle of curves representing the pairs belonging to the link. A joint J of node α is depicted as an arc convex towards α crossing the links involved in J . A joint involving just one link, as in (a) and (b), is called a *unitary joint*, and one containing several links, as in (c) and (d), is called a *multiple joint*. Distinct links involved in the same multiple joint on node α have disjoint α -domains. A joint is *partial* if it stretches out of the involved links, as in (b) and (d), and *total* otherwise, as in (a) and (c). The union of α -domains of the links involved in the same joint on node α is equal to α . A total unitary joint as in (a) is not explicitly shown as an arc. Partial joints on node α are *complementary* when the union of the α -domains of the links involved in them is α . Complementary joints are indicated by a dashed arc crossing these links. So the union of the α -domains of the three links is α in (e). When node α and β are connected by link δ and the joint of β involving

δ is total and unitary, α and δ are said to *dominate* β .

The initial structures of predicate domains are shown in Figure 3. Such structures, as

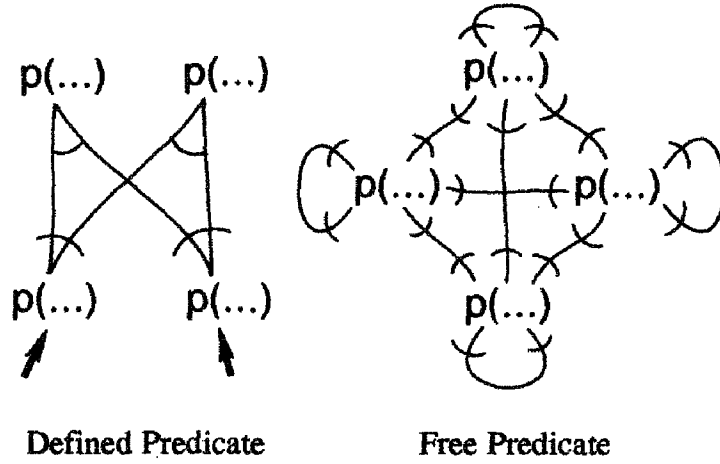


Figure 3: Predicate Domains

well as the other structures, will change as computation proceeds.

3 Computation

Here we introduce a method of symbolic computation together with some general control heuristics for controlling computation. There are two types of symbolic operation: *subsumption* and *deletion*. Here we chiefly concern ourselves with subsumption,

3.1 Subsumption

'Subsumption' means two things: *subsumption relation*, which we defined above, and *subsumption operation*, which we discuss below.

The purpose of a subsumption is to let information flow from a node. A node α may have *probes*. α is called the *origin* of these probes. Each probe is placed on a link and directed towards an endnode. The origin of a probe subsumes the node behind the probe. Probes transmit information of their origins across the network via subsumptions. The origin of probes has its *scope*. The scope of node α is the part \mathcal{S} of the constraint network satisfying the following conditions.

- \mathcal{S} is a connected graph containing α .
- A node β is behind a probe π on link δ and with origin α , iff β is in \mathcal{S} but δ is not.
- α subsumes every node in \mathcal{S} .

So the scope of α may be illustrated as in Figure 4, where arrows are probes, which just cover the boundary of the scope.

Every node α can just once create probes on all the links connected to α so that α is behind these probes. Subsumption extends the scope of a node by advancing probes, while preserving the instantiation possibilities of the network described above. We consider a

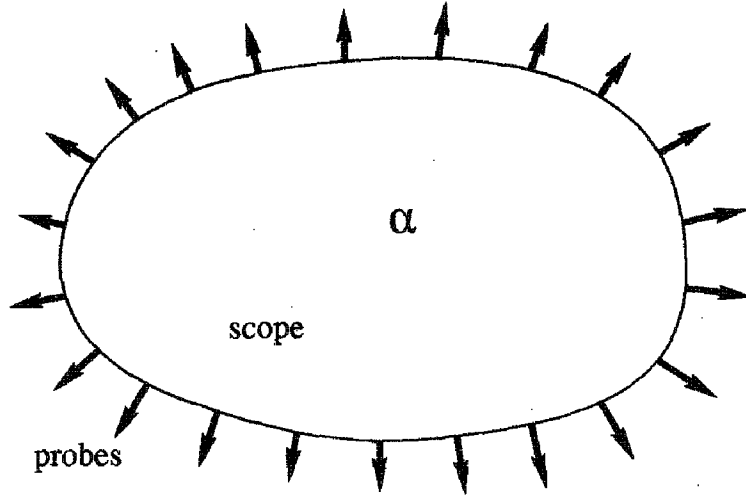


Figure 4: Scope of Node

subsumption from node ι to node ξ along link δ . ι , ξ , and δ are called the *input node*, the *target node*, and the *axis*, respectively, of this subsumption. The joint J of ξ involving δ is called the *target joint*. This subsumption extends the scopes of the origins of the probes on δ directed towards ξ . It proceeds as follows.

First, the set Π of the probes on δ towards ξ is detached from δ , and δ is shifted from J to another joint J' , as illustrated in Figure 5. J' is a copy of J and is on a node ξ'

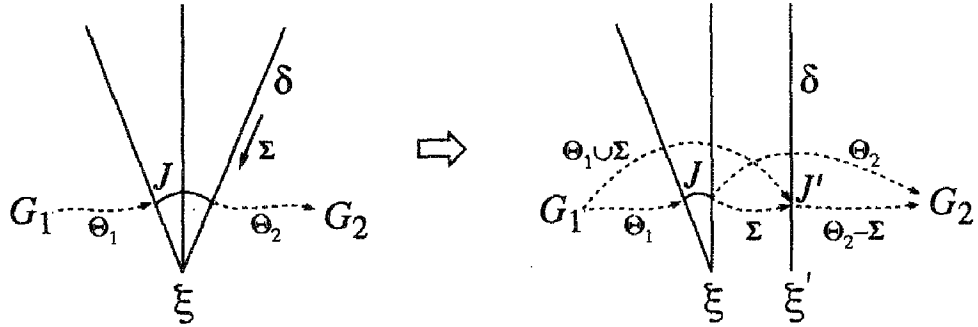


Figure 5: Shift of Link and Augmentation of Foldability

which is a copy of ξ . J' and ξ' may be created here and now, but may also have been made in a previous subsumption, as mentioned below. Below we proceed so as to make $\xi_0 = \xi_1 \cup \xi' \wedge \xi_1 \cap \xi' = \emptyset$ true, where ξ_0 and ξ_1 stand for ξ before and after this subsumption, respectively.

A joint may be *foldable* to another joint by a set of origins of probes. Each joint involved here, called a *foldable joint*, is one obtained by copying zero or more times a multiple joint in the initial state of computation. Typically, a foldable joint is one involving links in the predicate domain of a defined predicate. No joint just created is foldable to any joint. For any joint G and set Θ of nodes, there is at most one joint H such that G is foldable to H by Θ .

Let Σ be the set of origins of the probes in Π . If J is foldable, then for each joint G

the foldability relation extends in the following way, as illustrated in Figure 5, where the foldability relation is depicted by dashed arrows.

- J is foldable to J' by Σ .
- If G is foldable to J by Θ , then G is foldable to J' by $\Theta \cup \Sigma$.
- If J is foldable to G by Θ such that $\Theta \supset \Sigma$, then J' is foldable to G by $\Theta - \Sigma$.

If there has already been a joint to which J is foldable by Σ , then J' is that joint, ξ' is the node on J' , J' becomes a total multiple joint, and the foldability relation remains unchanged. Otherwise, J' and ξ' are newly created, δ dominates ξ' , and the foldability relation is augmented. We call the former case *folding*, and the latter *unfolding*.

If α is a proper atomic constraint or an argument of a proper atomic constraint, then $\hat{\alpha}$ stands for the set whose elements are this proper atomic constraint and its arguments; otherwise $\hat{\alpha} = \{\alpha\}$.

In the case of unfolding, each node ν in $\hat{\xi}$ is copied to ν' , and each link σ ($\sigma \neq \delta$) connecting ν and η is copied to σ' connecting ν' and some node η' . η' is the copy of η if $\eta \in \hat{\xi}$ and $\eta' = \eta$ otherwise. Relevant Joints are copied accordingly so as to preserve the instantiation possibilities of the network.

There are two cases, *splitting* and *non-splitting*, about how to create σ' . In the former, it is guaranteed that no layer of the splitting domain including σ before the copy overlaps with both ν and ν' after the copy. Such a guarantee is obtained iff $\sigma = R \cup R^{-1}$ for some bijection R or (inclusive) δ and σ belong to the same splitting domain. There is no such guarantee in the non-splitting case.

In the splitting case, as is illustrated in Figure 6, the η -domains of σ and σ' are disjoint

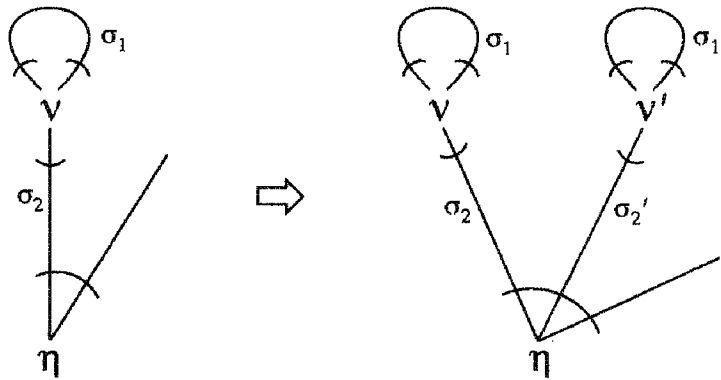


Figure 6: Copy of Links (Splitting)

when $\eta' = \eta$.

In the non-splitting case, as is illustrated in Figure 7, if σ was a loop, ν and ν' is connected by an additional link representing a relation pertaining to the layers overlapping both ν and ν' . Further if σ was involved in a multiple joint of η , then a subsumption along σ to η must be done before creating σ' ; otherwise the right instantiation possibilities cannot be represented.

In both splitting and non-splitting cases, the probes that ν had, if any, are deleted, and ν and ν' are licensed to generate new probes. Then every remaining probe on σ is copied to a probe on σ' , towards ν' , and the same origin. Further, each probe in Π is

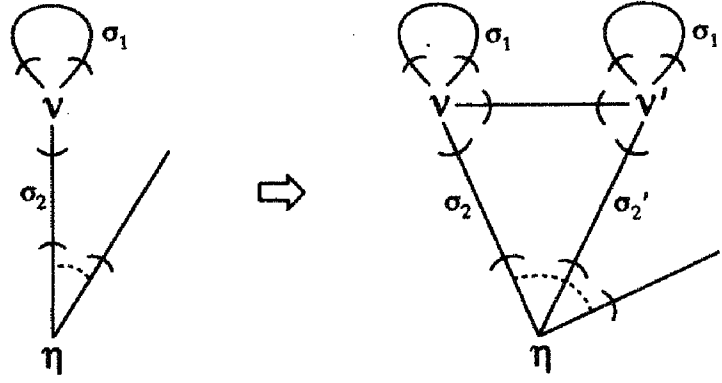


Figure 7: Copy of Links (Non-Splitting)

advanced through ξ' onto every link τ ($\neq \delta$) connected with ξ' so that ξ' should be behind the probe. If there is another probe on τ towards ξ' and with the same origin, then both probes are deleted.

Finally, in both folding and unfolding, if δ dominated ξ before this subsumption, ξ is deleted because it has become the empty set now. This deletion propagates across links and nodes until possibly non-empty sets are encountered: that is, until you come across partial or multiple joints of remaining nodes.¹ Now the subsumption is done.

To properly split splitting domains, we must augment this subsumption procedure so that a probe may carry, instead of origin, some information about which layers of the relevant splitting domain are involved in the node behind the probe. Such probes are transmitted from proper atomic constraints to their arguments and vice versa. A link is deleted if it contains two probes with opposite directions and associated with disjoint sets of layers. Further details are omitted due to the space limitation.

So far we have discussed subsumption in general. Below we describe the particularities of subsumptions along equalities and subsumptions along inference links.

A subsumption along an equality is triggered by a *dependency* between arguments. We say that there is a dependency between two arguments, when they *compete* with each other and are connected by a *dependency path*. Nodes α and β compete with each other when they are the first arguments of

- two bindings (as in $\xi=f(\bullet)$ and $\eta=g(\bullet)$),
- a binding and a feature specification, or
- two feature specifications with the same feature.

A dependency path connecting α and β is a sequence $\delta_1\delta_2\cdots\delta_n$ of *strong equalities* such that the endpoints of δ_i are α_{i-1} and α_i ($1 \leq i \leq n$), δ_i and δ_{i+1} are involved in different joints of α_i one of which is total ($1 \leq i < n$), $\alpha_0 = \alpha$ and $\alpha_n = \beta$. An equality is strong when it belongs to a clause or the predicate domain of a defined predicate, or when a subsumption has taken place along that equality.

¹This combination of copy and deletion is vacuous and thus may be omitted in actual implementation for the unfolding cases. The deletion of probes in the splitting case may also be avoided in such a situation.

A probe π on an equality δ might trigger a subsumption to advance π , when there is a dependency between the origin α of π and another node β and δ is included in a dependency path connecting α and β .

Suppose the scope of α includes another node β competing with α . If the proper atomic constraints A and B , each involving α and β as the first argument, respectively, are connected by an inference link δ , then δ *absorbs* B , as shown in Figure 8. That is,

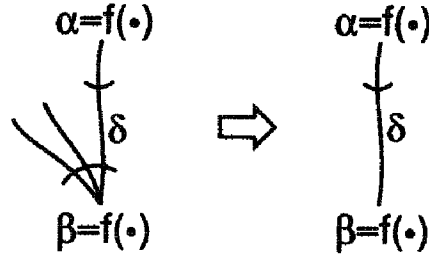


Figure 8: Absorption

the joint of B involving δ is modified so that β dominates β , because A has turned out to subsume B . Any other inference link involved in this joint is deleted, because it has turned out to be the empty set. Of course each equality accompanying δ must absorb its endnode in B at the same time. If there is no inference link between A and B , then B is deleted. Deletions of links and nodes propagate so long as the empty set is encountered, as said before.

A subsumption along an inference link may be triggered by *cost* assigned to the input node. Each literal in a clause may be assigned a cost. Similarly to the *assumability cost* of Hobbs et al. [5], the cost of a literal corresponds to the difficulty to abductively assume its negation. For instance, if you want to assume atomic constraint α by using a clause backwards whereas the cost of the literal $\neg\alpha$ in this clause is not zero, then you are to do something in order to cancel the cost. In this sense, an atomic constraint with a cost is regarded as a *goal* to achieve, and the abductive usage of the clause which gives rise to the goal is regarded as the *motivation* to set up that goal. A cost may be canceled by making the atomic constraint subsume another which is more readily believable. That is, a goal is fulfilled when it is established by some reason other than its motivation.

The input node of a subsumption along an inference link is the goal atomic constraint in the rest of the paper.² Such a subsumption eliminates the cost if the target node has been derived from the top clause without recourse to that very subsumption. Otherwise the cost is inherited into the clause which contains the output node. In a Horn clause normally used with all the atomic constraints therein being true, the head literal inherits the cost from a body atomic constraint, and the body atomic constraints inherit the cost from the head literal. We neglect the cost inheritance among body atomic constraints.

3.2 Heuristics

Subsumptions along equalities and those along inference links both encompass top-down and bottom-up information flow. Some heuristics are necessary to control such an oth-

²Subsumptions for checking consistency need not be triggered by cost.

erwise promiscuous system of computation so that more relevant pieces of information should be exploited with greater preferences.

Each heuristic for a subsumption along an equality is that one of the following conditions raises the preference of such a subsumption.

- (H1) The origin of a probe on the axis is close to (typically included in) the top clause or is a constant.
- (H2) A dependency path involving the axis and connecting an argument with the origin of a probe on the axis is short.

Both these conditions are regarded as indicating that the transmitted information (about the origin) is highly relevant to the destination of this transmission. In this connection, a subsumption along an equality is unlikely to happen if the axis belongs to the predicate domain of a free predicate and the target joint is partial, since the conveyed information would not be very relevant to the target node.

As for subsumptions along inference links, the following conditions each raise the preference.

- (H3) Corresponding arguments of the input node and the target node are connected via short dependency paths with the same node. (That is, those arguments are 'shared'.)
- (H4) The target node has already been derived from the top clause.

(H3) raises the possibility for instances of the two arguments to coincide in the domain of interpretation. (H3) amounts to a generalization (or relaxation) of the condition on which an inference link absorbs one of its endnodes. (H4) guarantees that the subsumption in question will lead to an immediate elimination of the cost of the input node. Probably (H4) could be relaxed to be a graded condition.

4 Parsing

Let us consider a simple case of context-free parsing based on the following grammar.

$$\begin{aligned} P &\rightarrow a \\ P &\rightarrow P P \end{aligned}$$

A parsing based on this grammar is formulated by the program as follows.

$$\begin{aligned} (5) \text{ true } &-p(A_0, B) \quad -A_0=[a|A_1] \quad -A_1=[a|A_2] \quad \dots; \\ &(\Phi) \quad p([a|X], X); \\ &(\Psi) \quad p(X, Z) \quad -p(X, Y) \quad -p(Y, Z); \end{aligned}$$

Depicted in Figure 9 are the four types of clauses created by this parsing process. A $\bullet=[a|\bullet]$ is a shorthand representation for a $\bullet=[\bullet|\bullet]$ plus an equality between the second argument and (the argument bound by) a . (a) is a copy of clause Φ in (5), and the other clauses are copies of Ψ . A label i of a link means that the relevant part of the network is in the scope of argument A_i . The reason why only these types of clauses are generated is that in this case every dependency arises between a $\bullet=[a|\bullet]$ in the top clause and

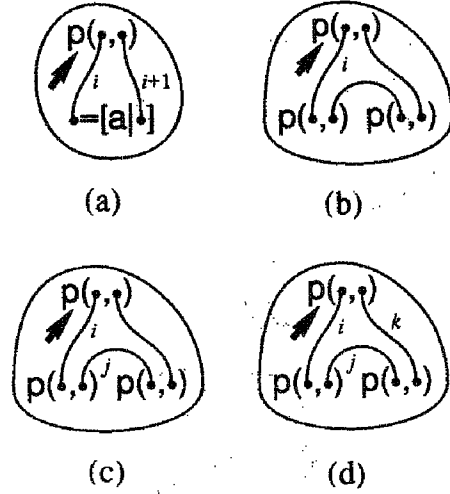


Figure 9: Clauses Produced through Parsing

another $\bullet=[a|\bullet]$ somewhere else and the first argument of the former is the origin of the subsumptions to resolve that dependency. A strict proof will be obtained by mathematical induction. Since the number of these clauses is $O(n^3)$ due to (d) and each of them may be generated in a constant time, the time complexity of the entire parsing is $O(n^3)$, where n is the sentence length. Each clause is guaranteed to be generated in a constant time, because each foldability test can be performed in a constant time, as discussed later. By employing a general optimization technique, we can eliminate the clauses of type (d), so that the space complexity is reduced to $O(n^2)$. Thus, our general control scheme naturally gives rise to standard parsing procedures such as Earley's algorithm and chart parsing.

(5) is graphically represented as Figure 10. We omit the links involved in the predicate

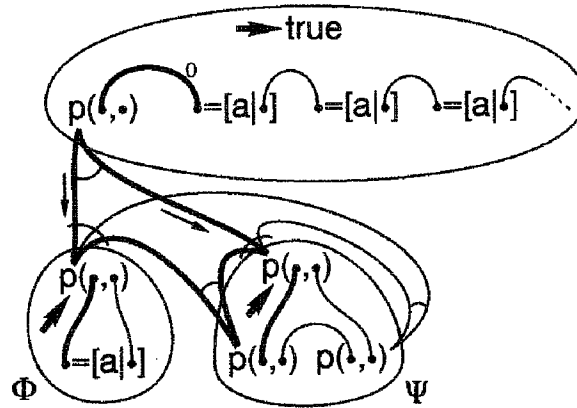


Figure 10: Parsing (1)

domain of a free predicate, until they are modified as in Figure 8. Thus no links among $\bullet=[a|\bullet]$ s are shown in Figure 10. Here is a dependency between the first $\bullet=[a|\bullet]$ in the top clause and the $\bullet=[a|\bullet]$ in Φ , as indicated by the dependency paths, which consist of thick links. To let information flow from the top clause following the above heuristic (H1), we are to do the two subsumptions indicated by the two thin arrows.

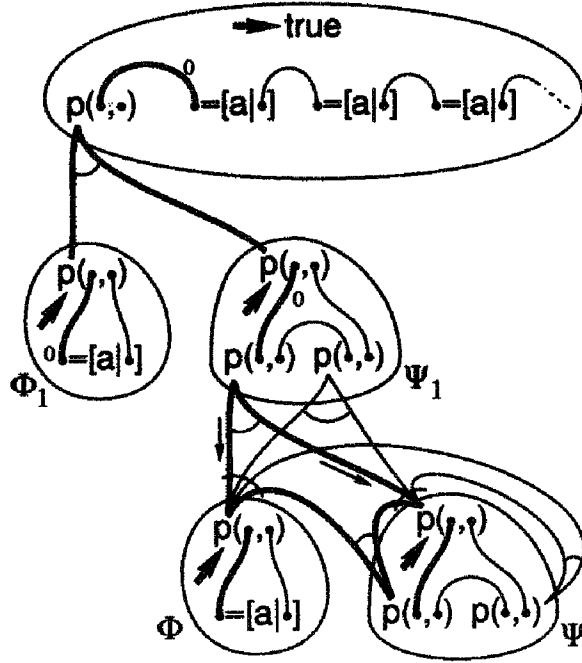


Figure 11: Parsing (2)

Those subsumptions copy Φ to Φ_1 and Ψ to Ψ_1 , resulting in Figure 11. For expository convenience, we assume here without loss of generality that copying of a clause produces a separate clause rather than one sharing atomic constraints with the original clause. Note that the first argument of the $\bullet=[a|\bullet]$ in Φ_1 is subsumed by A_0 .

Computation goes on into the same direction, and the two subsumptions are to happen as shown in Figure 11. Folding takes place this time, and the result is to shift the two inference links upwards, as in Figure 12. Now the first $\bullet=[a|\bullet]$ in the top clause dominates the $\bullet=[a|\bullet]$ in Φ_1 as indicated by the inference link between them, because, as indicated by number 0 in Φ_1 , the first argument of the former is within the scope of the first argument of the latter. Now the equality in the right-hand side of Φ_1 is within the scope of A_1 , as indicated in the figure. This subsumption also engenders a new set of dependencies between the first argument of the second $\bullet=[a|\bullet]$ in the top clause and that of $\bullet=[a|\bullet]$ in Φ , as indicated again by thick links in Figure 12. By executing the indicated subsumption following (H1), Ψ_1 is copied to Ψ_2 , so that we obtain Figure 13. Further advancing subsumptions as shown there, we get Figure 14. Computation goes on in the similar way.

As mentioned above, we are able to assume that each foldability test is performed in a constant time. This assumption is justified by, for instance, sorting the foldability information from each joint in the chronological order of the first subsumption which advanced probes with the relevant origin. In the present parsing example, this order happens to be the increasing order of the suffix i of A_i .

It is straightforward to integrate such a phrase-structure parsing with computation on internal structures of grammatical categories represented in terms of feature bundles, for instance. See [2, 4] for further details in this regard. Note that the above derivation of the parsing process is more general than the parsing-as-deduction approaches [6, 7], because it is free from stipulation of the left-to-right and to-down processing direction and also

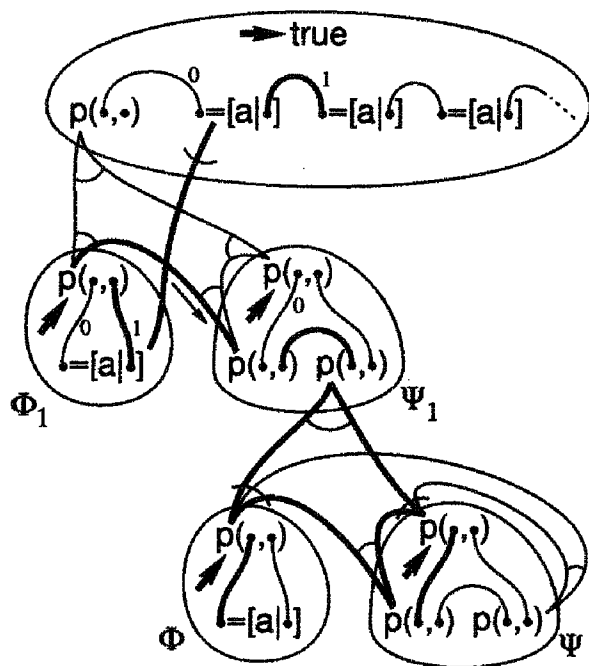


Figure 12: Parsing (3)

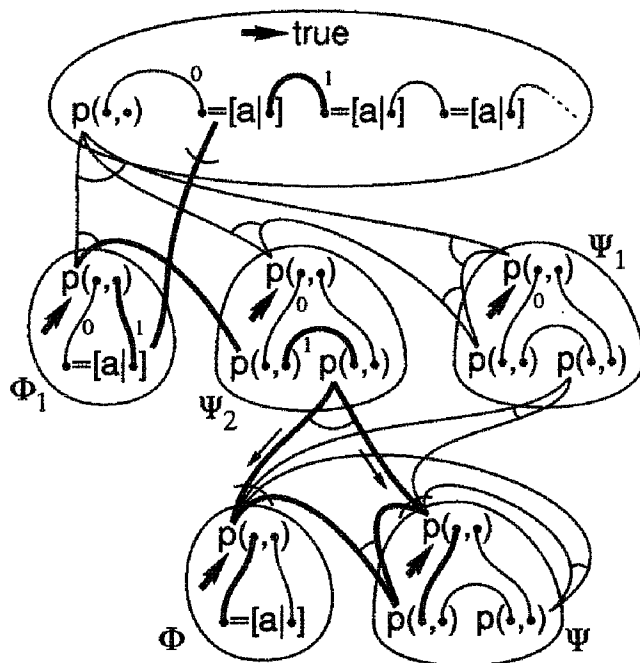


Figure 13: Parsing (4)

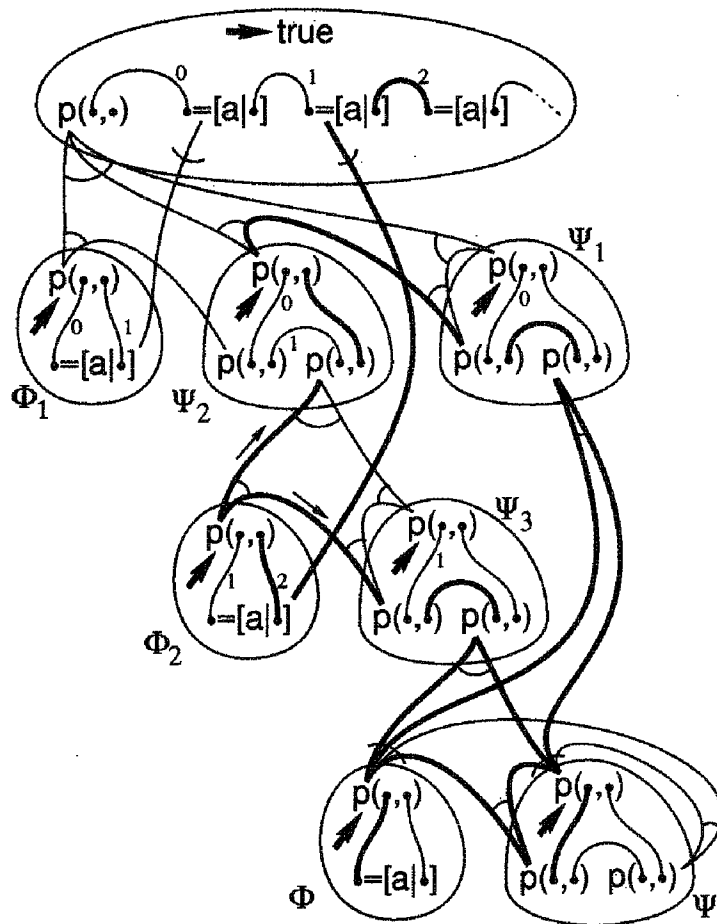


Figure 14: Parsing (5)

from task-dependency with regard to parsing or context-free grammar.

5 Generation

Here we consider how to verbalize the following semantic content in English.

$S \models \langle\langle \text{laughed}, \text{kim} \rangle\rangle$

This means that Kim laughed, based on Situation Theory [1]. That is, in some situation S there is an event which is of the sort `laughed` and whose agent is `kim`. So a sentence we might want to generate is 'Kim laughed.' S may be interpreted as, for instance, the speaker's model of the hearer's model of the world. A state of affairs $\langle\langle \text{laughed}, \text{kim} \rangle\rangle$ will be regarded as variable $L1$ constrained by two feature specifications $\text{rel}(L1, \text{laughed})$ and $\text{agt}(L1, \text{kim})$.

The initial state of computation could be formulated in terms of a program including the following clauses, among much more others.

- (A) $\text{true} \text{ -s}(\text{SEM}, W0, W1) \text{ -S} \models \text{SEM} \text{ -say}(W0)$
 $\text{ -S} \models L1^{\$}, \text{ -rel}(L1, \text{laughed})^{\$}$
 $\text{ -agt}(L1, \text{kim})^{\$} \dots;$
- (B) $\text{s}(\text{SEM}, X, Z) \text{ -np}(\text{SBJSEM}, X, Y)$
 $\text{ -vp}(\text{SEM}, \text{SBJSEM}, Y, Z);$
- (C) $\text{np}(\text{kim}, X, Y) \text{ -X} = [\text{'kim'} | Y]^{\$};$
- (D) $\text{vp}(L, \text{AGT}, X, Y) \text{ -X} = [\text{'laughed'} | Y]^{\$}$
 $\text{ -rel}(L, \text{laughed}) \text{ -agt}(L, \text{AGT});$

$\text{say}(W0)$ means that the utterance beginning at $W0$ should be actually uttered. $\text{S} \models \text{SEM}$ and $\text{S} \models L1$ separately exist in (A), because the next utterance need not directly refer to $L1$. For instance, one can mean that Kim laughed by saying 'Do you know that Kim laughed?' instead of just 'Kim laughed,' or by doing something other than utterance. One might even just give up the goal and say something quite different.

A '\$' attached to an atomic constraint represents a cost, so that the atomic constraint is a goal. The three goals in (A) altogether amount to a macroscopic goal to make a state of affairs $\langle\langle \text{laughed}, \text{kim} \rangle\rangle$ hold in situation S .

What we would like to demonstrate below is again that the control heuristics described in Section 3 tend to trigger the right operations depending upon the computational context, provided that the current goal is to be reached by some linguistic means; that is, by eventually uttering some sentence. Below we pay attention to only one maximal consistent structure of the sentence at a time just for the sake of simplicity, but the actual generation process may involve OR-parallel computation similar to that in parsing of the previous section.

Figure 15 graphically represents clauses (A) and (C). A proper atomic constraint with a binary predicate, possibly together with equalities involving the two arguments, is represented here as an arrow from (an argument equalized with) the first argument to (an argument equalized with) the second argument. Links in predicate domains are selectively displayed for expository simplicity.

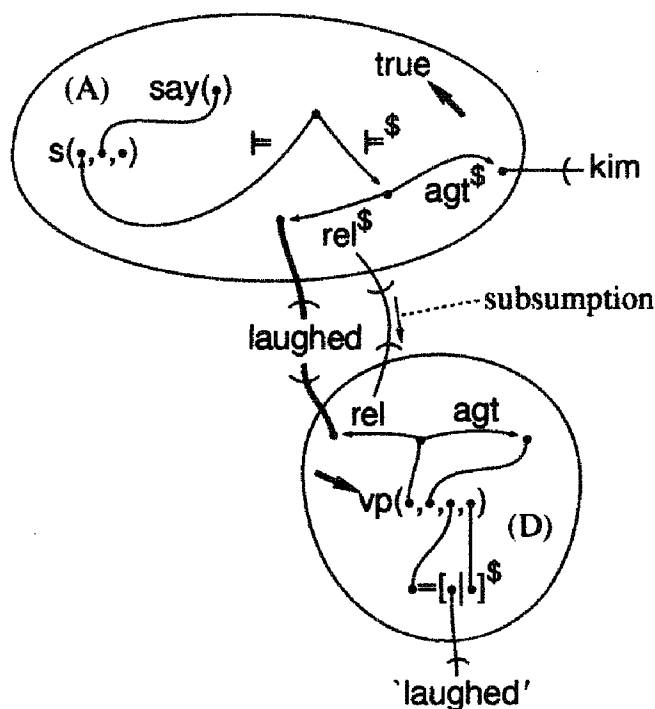


Figure 15: Generation (1)

The most probable operations to take place here are subsumptions involving one of these three goals. There should be innumerable combinations for such subsumptions, because the speaker's lexicon must include a large number of atomic constraints of the form $\bullet \models \bullet$, $\text{rel}(\bullet, \bullet)$ and $\text{agt}(\bullet, \bullet)$, even though subsumptions with extralinguistic parts of the constraints are excluded due to the above provision that the current goal is to be fulfilled by linguistic means.

However, two of such subsumptions are preferred to the others. One is the subsumption concerning the two $\bullet \models \bullet$ s in (A), and the other is from the $\text{rel}(\bullet, \bullet)$ in (A) to that in (D). In both cases, the two atomic constraints share the same argument for the same argument place, which raises the preference due to (H3). Let us tentatively choose just the latter subsumption in this particular presentation. No big difference would follow in the long run, even if the former subsumption or both were chosen instead.

By the subsumption concerning the two $\text{rel}(\bullet, \bullet)$ s, we obtain the structure shown in Figure 16. We have copied clause (D) to (D'), because the $\text{rel}(\bullet, \bullet)$ in (A) is a goal. Now in Figure 16, $\text{vp}(\bullet, \bullet, \bullet, \bullet)$ in (D') is a goal, by inheriting the cost from $\text{rel}(\bullet, \bullet)$ of (A). The cost of $\bullet = [\bullet | \bullet]$ in (D') is inherent, as indicated in (D). Now the most probable next computation is the sequence of subsumptions along the thick link(s) constituting a dependency path. Following the heuristic (H1), those subsumptions propagate from the top clause. After that, the inference link between the two $\text{agt}(\bullet, \bullet)$ s absorbs the one in (B).

This gives us Figure 17. (D') has not been duplicated here, because the above subsumptions did not actually duplicate any clause. In this context, the subsumption concerning the two $\text{vp}(\bullet, \bullet, \bullet, \bullet)$ s is possible, since the one in (D') is a goal. Due to (H3), this subsumption is more preferable than the others concerning two $\text{vp}(\bullet, \bullet, \bullet, \bullet)$ s, because their first arguments are both connected to **kim** (that is, the first argument of $\bullet = \text{kim}$) via short dependency paths. As a result, (B) is copied to (B') and the $\text{vp}(\bullet, \bullet, \bullet, \bullet)$ in (B')

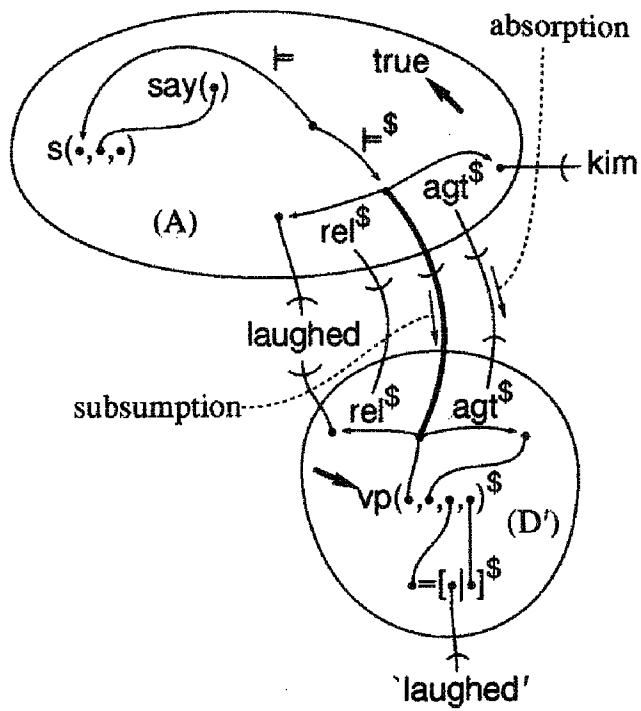


Figure 16: Generation (2)

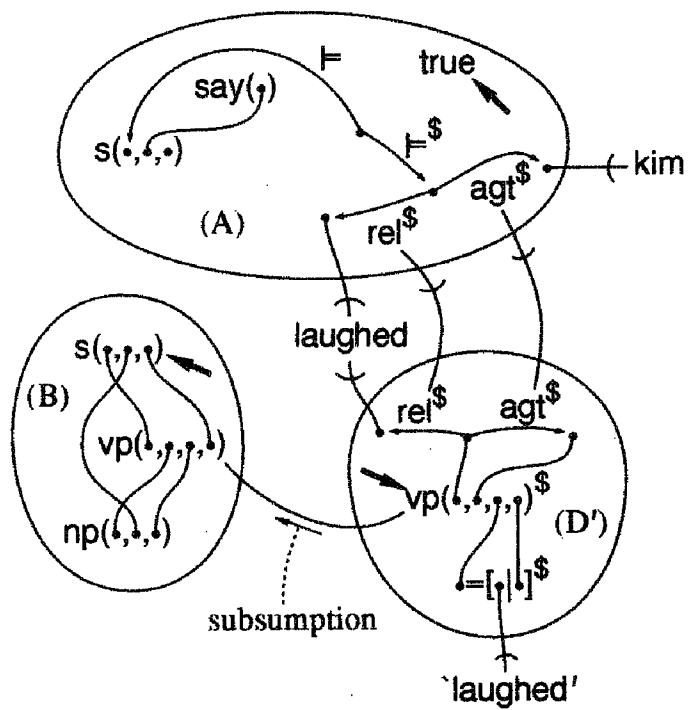


Figure 17: Generation (3)

is dominated by that in (D'), as in Figure 18.

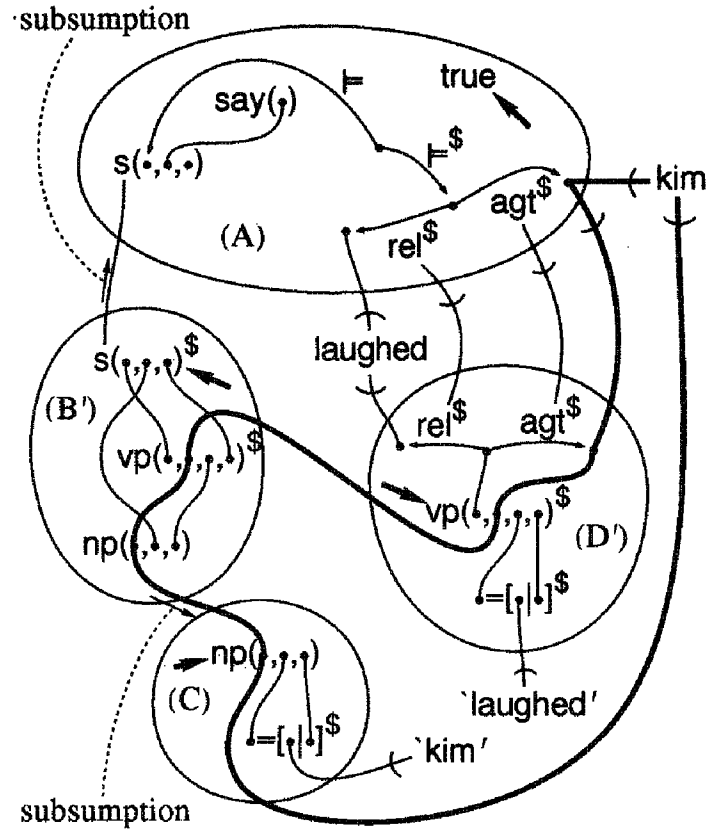


Figure 18: Generation (4)

Now that $s(\cdot, \cdot, \cdot)$ in (B') is a new goal, it is caused to subsume another $s(\cdot, \cdot, \cdot)$ in (A). According to (H4), this subsumption is particularly preferable because (A) is the top clause. On the other hand, the subsumption from the first argument of $np(\cdot, \cdot, \cdot)$ in (B') to the first argument of $np(\cdot, \cdot, \cdot)$ in (C) could take place here, to resolve the cyclic dependency about kim referred to from (A) and (C). This subsumption is the most probable operation concerning this dependency in this context, because it is along the shortest relevant dependency path. We assume that the direction of this subsumption is downwards, as indicated in Figure 18. It will be the same in the long run if it were in the opposite direction.

The mentioned computation in Figure 18 takes us to Figure 19. We have a new top clause (A'), which shares most part of itself with (A), except the copy of $s(\cdot, \cdot, \cdot)$. Some of the previous goals have disappeared due to the subsumption concerning $s(\cdot, \cdot, \cdot)$ s. Now the remaining goals are $\bullet = [\bullet | \bullet]$ s in (C') and (D') and the $\bullet \models \bullet$ in the intersection of (A) and (C'). We might do a subsumption concerning the two $\bullet \models \bullet$ s, because they share both the arguments. This subsumption could have happened earlier, of course, particularly ever since both arguments came to be shared in Figure 16 via (B) and (D'). As mentioned before, however, it would have caused no essential difference eventually. At the same time we could execute the procedure $\text{say}(\bullet)$ to realize the goal $\bullet = [\bullet | \bullet]$ in (C'). It is reasonable to assume that this computation is triggered by the fact that the argument of $\text{say}(\bullet)$ subsumes the first argument of this $\bullet = [\bullet | \bullet]$. This heuristic for firing procedures looks generally applicable not only to utterance but also to every other output

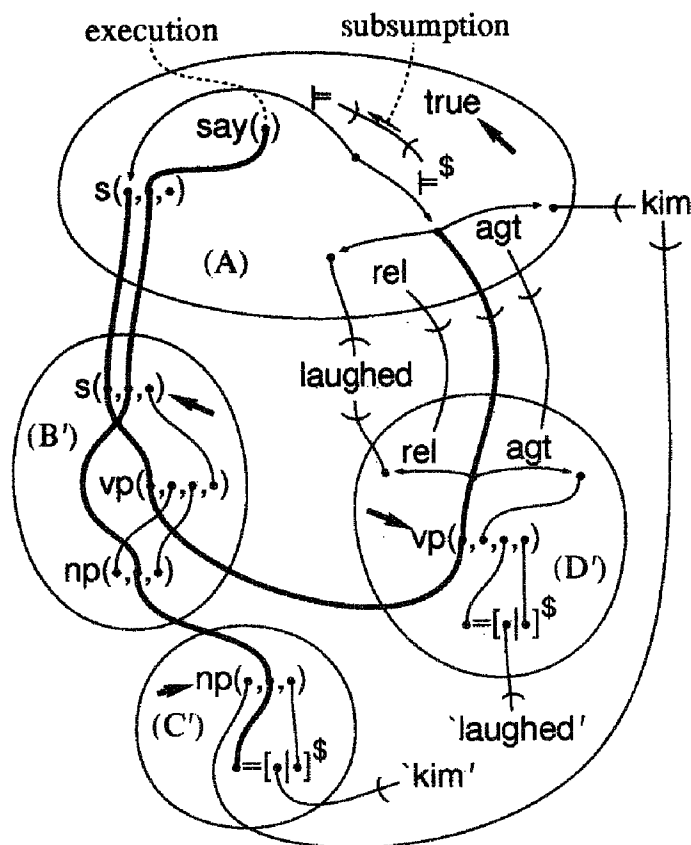


Figure 19: Generation (5)

procedure.

Thus we move to a new computational context in Figure 20. The execution of $\text{say}(\bullet)$ has created a new $\bullet=[\bullet|\bullet]$, so that 'Kim' has been spoken aloud. This $\bullet=[\bullet|\bullet]$ dominates the $\bullet=[\bullet|\bullet]$ in (C'), as indicated by the thick link. Generation of 'Kim laughed' completes if $\text{say}(\bullet)$ is executed one step further.

Note that this generation process captures the bottom-up feature of semantic head-driven generation [8], especially when we move from Figure 15 through Figure 18. The subsumption concerning the arguments of $\text{np}(\bullet, \bullet, \bullet)$ s happening between Figure 18 and Figure 19 captures the top-down aspect as well.

6 Concluding Remarks

We have introduced a set of general heuristics for controlling symbolic computation on logic constraints, and demonstrated that sentence parsing and generation are attributed to these heuristics. In the above presentation, parsing is for the most part based on truth maintenance (resolution of dependencies among arguments) controlled by heuristics (H1) and (H2), whereas generation is more dependent on goal satisfaction controlled by (H3) and (H4). In more realistic cases, however, both processes would involve both kinds of computation in a more intertwined way.

A related nice feature of our framework is that, in principle, all the types of constraints — syntactic, semantic, pragmatic and extralinguistic — are treated uniformly and integrated naturally, though a really efficient implementation of such an integrated

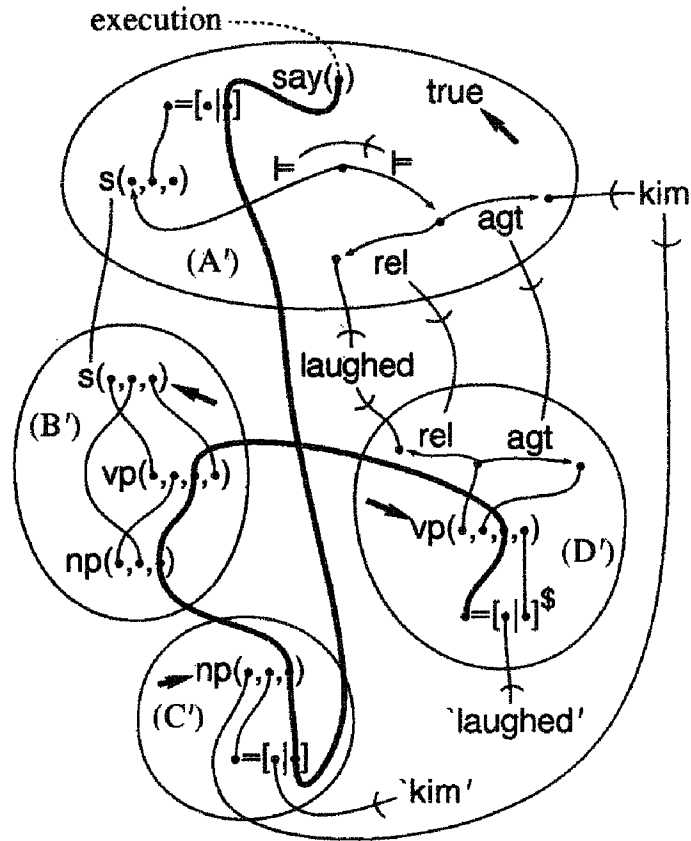


Figure 20: Generation (6)

system requires further research. In this connection, we have undertaken to study how to implement the above heuristics in a more principled and flexible way, based on a notion of *potential energy* [3], but the present paper lacks the space for discussing the details.

In this paper we have discussed only task-independent aspects of control heuristics. Our conjecture is that we will be able to dispense with domain-dependent and task-dependent control heuristics altogether. The domain/task-dependent characteristics of information processing will be captured in terms of the assignment of energy functions to the relevant constraints. The resulting system will still be free from stipulation of the directions of information flow, allowing multi-directional information processing, since neither the symbolic component nor the analog component (that is, energy) of the constraint refers explicitly to information flow.

References

- [1] Barwise, J. (1990) *The Situation in Logic*, CSLI Lecture Notes No. 17.
- [2] Hasida, K. (1990) 'Sentence Processing as Constraint Transformation,' *Proceedings of ECAI '90*, pp. 339-344.
- [3] Hasida, Kôiti (1992) 'Dynamics of Symbol Systems: An Integrated Architecture of Cognition,' to appear in *FGCS '92*.
- [4] Hasida, K. and Tsuda, H. (1991) 'Parsing without Parser,' *International Workshop on Parsing Technologies*, pp. 1-10, Cancun.

- [5] Hobbs, J., Stickel, M., Martin, P., and Edwards, D. (1988) 'Interpretation as Abduction,' *Proceedings of the 26th Annual Meeting of ACL*, pp. 95-103.
- [6] Pereira, F. C. N. and Warren, D. H. D. (1983) 'Parsing as Deduction,' *Proceedings of the 21st Annual Meeting of ACL*, pp. 137-144.
- [7] Shieber, S. M. (1988) 'A Uniform Architecture for Parsing and Generation,' *Proceedings of the 12th COLING*, pp. 614-619.
- [8] Shieber, S. M., van Noord, G., and Moore, R. C. (1989) 'A Semantic-Head-Driven Generation Algorithm for Unification-Based Formalisms,' *Proceedings of the 27th Annual Meeting of ACL*, pp. 7-17.

J. Acoust. Soc. Jpn. (E) 12, 1 (1991)

Formant frequency extraction by moment calculation of speech spectrum

Keywords: Formant, Spectrum, Moment, Speech, Analysis

PACS number: 43.72.Ar

K. Takagi* and S. Itahashi**

*IBM Japan Ltd.,

1623-14 Shimotsuruma, Yamato, 242 Japan

**Institute of Information Sciences and Electronics,
University of Tsukuba,

1-1-1 Tennodai, Tsukuba, 305 Japan

(Received 10 April 1990)

1. Introduction

Formant frequency is one of the important parameters of speech spectra, and various kinds of automatic estimation methods have been studied for many years. The spectrum moment method utilizes only the gross features of a spectrum envelope.^{1,2)} With this method, the spectrum pattern is divided into several frequency regions, each of which has only one formant peak, then the centroid of each region is calculated to give an estimate of the formant frequency. This paper proposes an adaptive algorithm of frequency region division by repeating the region adjustment utilizing second-order and third-order spectrum moments not have been used in previous studies.³⁾

2. Calculation of Spectrum Moment

The k -th-order spectrum moment M_k of a speech spectrum is defined as follows.

M_k : the k -th-order spectrum moment

$F(i)$: center frequency of the i -th component of a spectrum

$A(i)$: power of the i -th component

N : number of the spectrum components

$$M_0 = \sum_{i=1}^N A(i) \quad (1)$$

$$M_1 = \sum_{i=1}^N A(i)F(i)/M_0 \quad (2)$$

$$M_k = \sum_{i=1}^N A(i)(F(i)-M_1)^k/M_0 \quad (3)$$

The first-order moment M_1 is the frequency centroid of the spectrum. The second-order and the third-order moments, M_2 , M_3 , reflect the variance and the skewness of the spectrum pattern, respectively.

The maximum and the minimum power of the spectrum is modified to be 1.0 and 0.0, respectively in order to guarantee positive, non-zero M_0 in calculation. Then

the constant bias of 0.1 (10% of the spectrum power dynamic range) is added to reduce the estimation sensitivity to speaker-dependent spectrum variations.

First-order spectrum moments divide a mel-scaled power spectrum as illustrated in Fig. 1. The M_1 of the region (0, MAX), FC, is calculated first, and the M_1 's of the lower half and the higher half of the region, FLC, FHC. The M_1 's of the three regions, (FLC, FC), (FC, FHC) and (FHC, MAX) are computed to give boundaries, FLCC, FHCC and FHHC. The M_1 of the three regions, (0, FLCC), (FLCC, FHC) and (FHCC, FHHC) are primary estimates of the first, second and third formant frequencies, $f_1(0)$, $f_2(0)$, $f_3(0)$, respectively.

3. Adjustment of the Frequency Regions

When the spectrum of a region is not symmetric the primary estimate does not give the exact frequency of the formant. The second-order and the third-order moments, M_2 , M_3 , are utilized to move the estimate closer to the peak by adjusting the frequency region. The formant peak lies on the lower frequency side of M_1 when $M_3 < 0$, and on the higher side when $M_3 > 0$. Therefore an adjustment algorithm has been devised to shift the boundaries of the region to the side where the peak exists in accordance with the sign of M_3 . The amount of frequency region shift is determined by M_2 , since a relatively large shift is required for a peak with a broad bandwidth, and a small shift is enough when the bandwidth is small (See Fig. 2). That is to say,

when $M_3 > 0$,

$$\begin{aligned} \text{modified region} = & (f_i(k-1) \\ & - C\sqrt{M_2}, f_i(k-1) + \sqrt{M_2}) \end{aligned} \quad (4)$$

when $M_3 < 0$,

$$\begin{aligned} \text{modified region} = & (f_i(k-1) \\ & - \sqrt{M_2}, f_i(k-1) + C\sqrt{M_2}) \end{aligned} \quad (5)$$

where $C=1.5$ (experimental value), $k=1, 2, 3, \dots$

Then let the M_1 's of the modified regions be the next estimates, $f_1(k)$, $f_2(k)$, $f_3(k)$. The width of each region becomes narrower as a result of the adjustment, but the minimum width is set at 262.5 mel, which corresponds to the width of 26 components of a 256-component spectrum. It is assumed that two adjacent formants will meet, but the trajectories will never cross.

An estimate comes closer to the formant as a result of region adjustments, then finally it goes back across the formant peak (See Fig. 3). Therefore a convergence condition is defined as follows.

Let $D_i(k)$ denote the difference between the k -th and $(k-1)$ -th estimates of the i -th formant,

$$D_i(k) = f_i(k) - f_i(k-1) \quad (6)$$

where $i=1, 2, 3$, and $k=1, 2, 3, \dots$ When Eq. (7) holds,

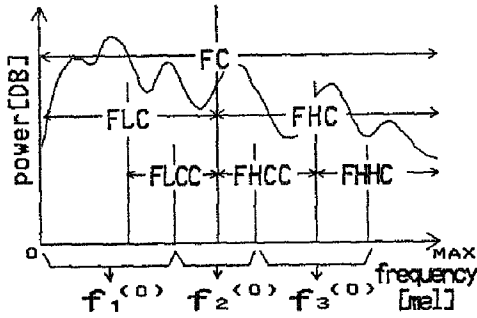


Fig. 1 Division of the frequency region.

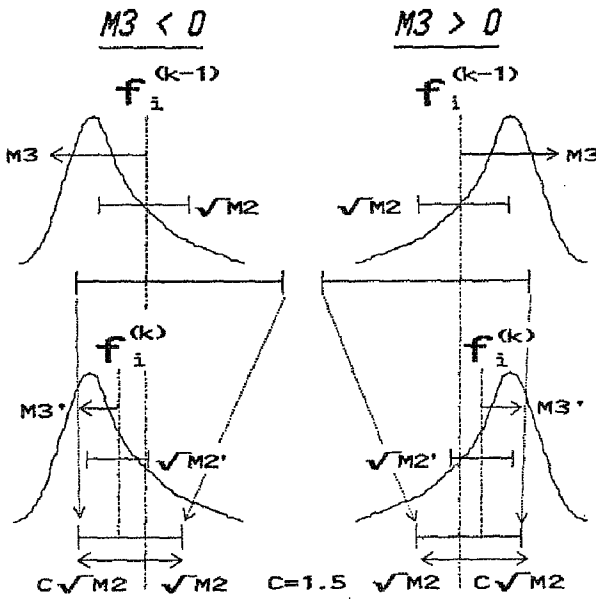


Fig. 2 Frequency region adjustment.

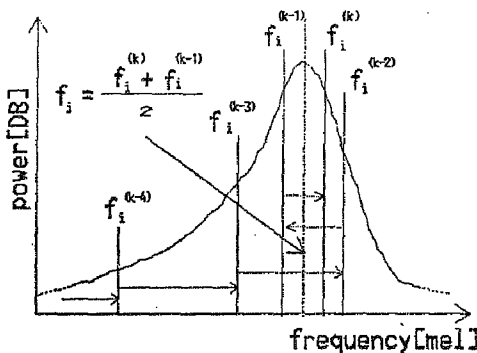


Fig. 3 Judgement of estimation convergence.

$$\begin{aligned} \text{sgn}(D_i(k-1)) &\neq \text{sgn}(D_i(k-2)) \\ &\wedge \text{sgn}(D_i(k)) \neq \text{sgn}(D_i(k-1)) \end{aligned} \quad (7)$$

where $\text{sgn}(x)$: the sign of x ,

let f_i be the estimate of the i -th formant frequency.

$$f_i = (f_i(k) + f_i(k-1))/2 \quad (8)$$

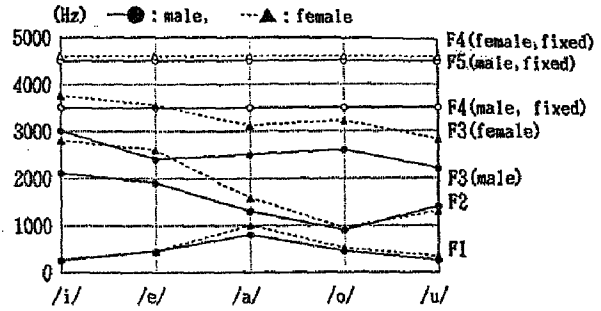


Fig. 4 Formant frequencies of the synthetic spectra of five Japanese vowels.

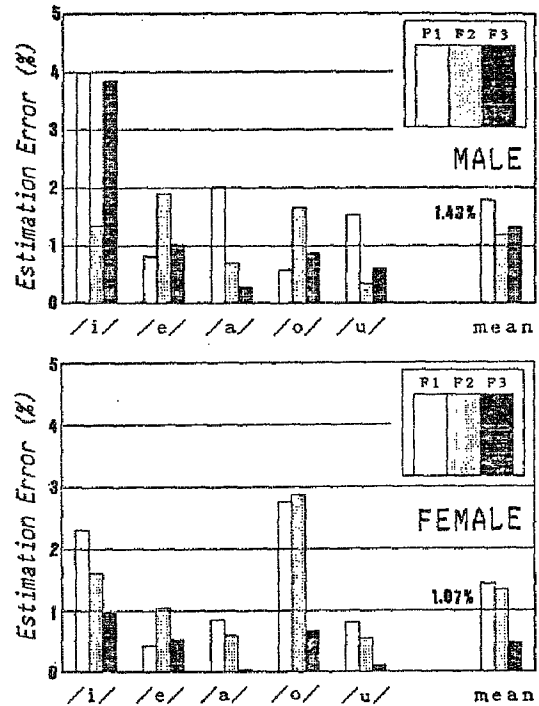


Fig. 5 Formant estimation error of the synthetic spectra.

4. Error Evaluation by Synthetic Spectra

The synthetic spectra of the five Japanese vowels were used for error evaluation. The spectra were produced by an all-pole model of speech spectrum. Typical formant frequencies of a male and a female voice are given as in Fig. 4, and the bandwidth B_w are given as a function of the formant frequency F .

$$B_w = 50(1 + F^2 \times 10^{-6}/6) \quad (9)$$

The estimation error was evaluated by the relative error between the true formant frequency F_i and its estimate f_i ,

$$E_i = |f_i - F_i|/F_i \times 100 [\%] \quad (10)$$

As shown in Fig. 5, the mean estimation error of the three formant frequencies of each vowel is 1.43% (min).

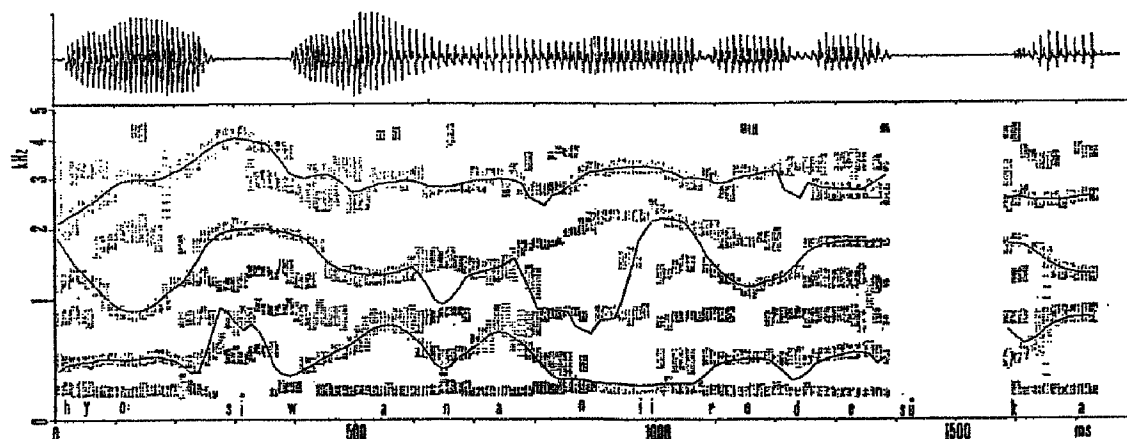


Fig. 6 Example of formant estimation of male speech /hyo:siwananiirodesuka/ (What color is the book cover?).

0.28%, max. 3.98%) for male spectra and 1.07% (min. 0.04%, max. 2.87%) for female spectra.

The estimation error mentioned above does not reflect the influence of excitation source; the data indicate a sort of upper limit of estimation precision. More realistic estimation would be possible with synthesized speech. However, even the experiment with synthetic speech does not reflect the real situation properly; final estimation should be performed with natural speech as described in the next section.

5. Analysis of Natural Speech

Speech material spoken by five male and five female speakers was analyzed. The material consists of 19 kinds of utterances, i.e. 5 isolated Japanese vowels, 3 kinds of connected utterances of vowels and 11 short interrogative sentences. The mean value of the neighboring five frames is taken as the estimate of each frame for smoothing.

Figure 6 shows an example of the results of the analysis (10 kHz sampling, 12 bits, 5 kHz LPF, 20-ms Hanning window, 10 ms frame interval, 256-point FFT spectrum envelope, cepstrum lifter: 16). The estimates lie close to most of the visually determined formant peaks in the analyzed interval, though there are errors in several frames.

The estimation error was evaluated by comparing the estimates of the spectrum moment method and the roots of the LPC equation. Estimates from the LPC equation were corrected visually and taken as the reference formant frequencies. The predictor order of the LPC analysis was set at 13.

The mean estimation error by the moment method of the five speakers are plotted in Fig. 7. The error for the vowel materials is 3.4% (min. 0.4%, max. 5.9%) for the male and 5.6% (min. 0.8%, max. 11.7%) for the female speakers. The error was larger for the interrogative sentences: 5.5% (min. 2.5%, max. 7.5%) for the male and 9.0% (min. 5.8%, max. 13.3%) for the female

Formant Estimation Error of the Real Speech Spectra
by the Spectrum Moment Calculation

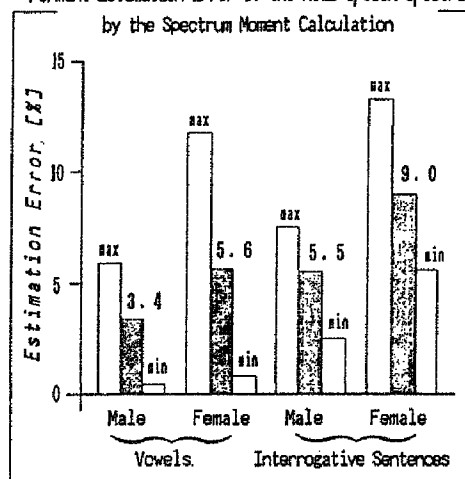


Fig. 7 Formant estimation error of the real speech.

speech.

6. Conclusion

A revised algorithm of spectrum moment method for formant frequency estimation was proposed in which second and third order spectrum moments are utilized to adjust the frequency regions for centroid calculation. The method proved to be effective for reducing errors of formant frequency estimation.

References

- 1) J. Suzuki, Y. Kadokawa and K. Nakata, "Extraction of formant frequencies by moment calculation," J. Acoust. Soc. Jpn. (J) 19, 106-114 (1963) (in Japanese).
- 2) M. Nakatsui and J. Suzuki, "Formant frequency estimation by inverse filtering and moment calcu-

lation and its evaluation using synthetic speech sound," *J. Acoust. Soc. Jpn. (J)* **26**, 211-221 (1970) (in Japanese).

- 3) S. Itahashi and K. Takagi, "Automatic formant frequency extraction by moment calculation of speech spectrum," *Proc. Eurospeech 89*, 207-210 (1989).



Kazuyuki Takagi received the B.S. and M.S. degrees in Information Science from University of Tsukuba, Tsukuba, Japan in 1987 and in 1989, respectively. He is currently working on application software development as an Associate Software Engineer at Asian Pacific Products Division of

IBM Japan Ltd.



Shuichi Itahashi received the B.S., M.S. and Ph.D. degrees in electrical communication engineering from Tohoku University, Sendai, Japan, in 1964, 1967, and 1971, respectively. From 1970 to June 1972 he was a Research Associate at the Research Institute of Electrical Communica-

tion, Tohoku University, where he worked on spoken word recognition utilizing linguistic information. In July 1972 he joined the research staff of the Electro-technical Laboratory, Ministry of International Trade and Industry, Japan, where he worked on speech analysis and perception. From 1977 to 1978 he was a guest researcher at the Royal Institute of Technology, Stockholm. In 1982 he moved to the University of Tsukuba, Tsukuba, Japan. Since then he has been working on speech, image and natural language processing. He has been chairman of the Speech Input Systems Subcommittee of the Japan Electronic Industry Development Association (JEIDA) since 1981, where he has been working with speech databases. Dr. Itahashi is a member of the Institute of Electrical and Electronics Engineers, the Acoustical Society of America, the Institute of Electronics, Information and Communication Engineers, the Information Processing Society of Japan, the Japanese Society for Artificial Intelligence and the Japanese Cognitive Science Society.

相互主導型対話機能を持つ音声応答システムとその対話管理手法

Speech Response System with Interruptive Dialogue
and its Method of Dialogue Management浅野 康治 広瀬 啓吉
Yasuharu ASANO Keikichi HIROSE東京大学 工学部
〒113 東京都文京区本郷7-3-1Faculty of Engineering, University of Tokyo
7-3-1 Hongo, Bunkyo-ku, Tokyo 113

あらまし 音声言語を用いてユーザに情報提供を行なう音声応答システムについて、その構成と対話管理手法について概説する。システムはユーザとの対話の履歴を管理し、応答内容を深層意味表現の形で生成する対話管理部と、その生成された深層意味表現をもとに表層文を生成し、音声を合成する文音声合成部の2つに大きく分けられる。対話管理部では処理に用いる規則を、質問応答型対話一般に対して成り立つ対話履歴の管理などに関する規則と、応答内容立案の際に必要な対話の話題に依存する規則、データベース検索などのタスクに依存する規則の3つに分けて構成しており、これによって対話の話題や応答情報を提供する知識ベースが変更された場合にも、システムの変更箇所を明確にすることが可能である。

Abstract A proposal is given for the construction of speech response system where the information is conveyed through spoken language. A method is also proposed for the dialogue management. This system consists of two parts; dialogue management part and speech synthesis part. The former manages the dialogue records and generates the deep structure for response, while the latter generates the surface sentences from the deep structure and synthesize the spoken sentences. For the dialogue management part, three sets of rules are prepared, i.e., one set of general rules and two sets of specific rules. General rules manage the dialogue records on questions and answers between the system and the user. One set of specific rules relates to the dialogue topics, while the other set relates to the tasks for the system. This classification of the rules enables one to clarify easily where to be modified in the system when the knowledge base is changed.

1. はじめに

計算機を用いた知的システムとそのユーザとの間のインターフェースを考えると、その最も有効な手段の1つは、人間同士の情報伝達における主な手段である自然言語、そのなかでも特に人間に対する負担の小さい音声言語を用いることであろう。このような観点から音声による知的システムとの対話を目指した研究が進められている。このようなシステムを考えた場合、ユーザに適切に応答するためには、システムは対話の流れを適切に保存・管理し、この管理した情報も考慮して応答する文を生成し、音声と

して提示する必要がある。

知的システムで直接処理し得る知識表現から音声合成を行なう研究は、1979年の Fallside⁽¹⁾以来いくつかの例がある。またこれらとは別に人間同士の対話を理解するという観点からの研究も行なわれており Litman⁽²⁾、飯田⁽³⁾の例があるが、これらはユーザに対する応答の立案という点からは力点が異なっている。

本研究では、ユーザとの間で双方向に情報伝達を行なうシステムの1つである質問・応答システムを取り上げ、対話を通じてユーザが必要とする情報を適切な自然言語で表現し、連続音

声の形でユーザに提示するシステムを構築することを目的とする。しかし単なるQAシステムのようにユーザからの質問にシステムが単純に答えるというようなものではなく、必要に応じてシステムがユーザに対して質問や提案を行ない、お互いが主導権を取り合うような対話を行い得るシステムの構築を目指している。これによってシステムは処理の途中で必要となった情報を収集し、ユーザにとってより適切な情報を提供することが可能となり、よりUser friendly なシステムを構築することができると思う。

本報では、このような日本語による音声応答システムについて、全体構成の概要と、その重要な機能の一つである対話管理に関して、これまでにやってきた研究内容を述べる。

2. システムの全体構成

システムの全体構成の概略を図1に示す。応答システムを構成するにあたっては、当然のことながら質問応答の話題となる分野に関する知識を提供し得るエキスパートシステムやデータベースなどの知識ベースが必要となる。本研究においては、知識ベースとして次のような理由から一般に提供されているスキーマに関するデータベースを用い、質問応答の内容としてはスキーマの案内及び紹介とした。

- ・対象分野に関する対話において用いられる語彙の量が数百個程度であり、小規模の単語辞書を作成することでシステムの性能評価が可能となる。
- ・データベースがテキストデータ、数値データ、二値データをそれぞれ含んでおり、データの種類のバラエティに富む。
- ・本システム専用にはデータベースを作るのではなく、一般に提供されているものを用いることによって、データベースの種類によってシステ

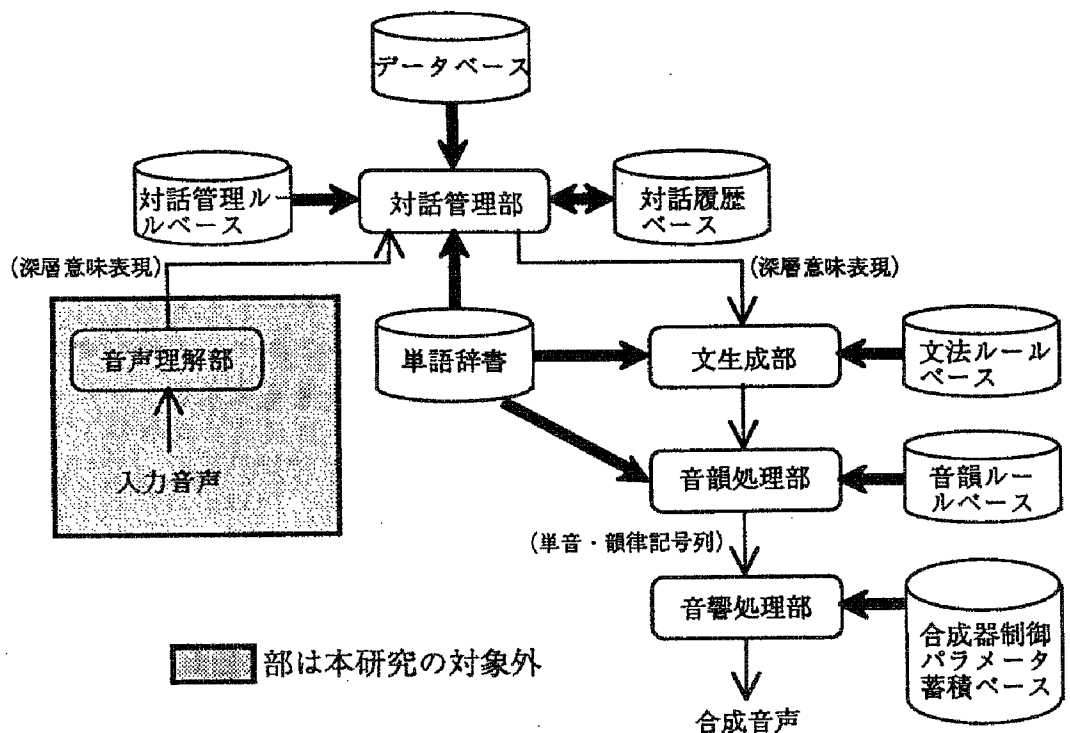


図1 システムの全体構成

ムに変更を要する部分を明確にし、さらにその変更のプロセスを明らかにすることによって、他のデータベースに対してもこのシステムの構成やアルゴリズムが有効であることを示すことが可能である。

以下でシステムを構成する各要素について概説する。

データベース (DB) :

前記の通り本研究においてはデータベースとしてスキーマに関するものを用いることとした。これには各スキーマに関して表1に示すようなデータ項目が登録されている。

単語辞書:

話題によって、対話において出現する単語は当然変わってくる。そこでシステムにおいて利用するデータベースに応じて、使用する語彙を登録した単語辞書を用意することが考えられる。辞書中において登録すべき情報は単語の品詞によって違い、以下のようにまとめることができる。

- ・全ての単語において登録する情報
 - 単語表記、品詞、音素表記、アクセント型
- ・動詞、形容詞などの述語において追加して登録する情報
 - 活用型
 - 深層格、表層格、必須・選択格の別

表1 データベースに各スキー場ごとに登録されているデータ項目(一部)

数値データ	コース数	リフト数	リフト料金	駐車場台数
テキストデータ	所在地	問い合わせ先	マイカー道	鉄道交通
二値データ	ナイターあり	ボール練習可	託児施設あり	温泉あり

データベースの検索条件との対応関係

・名詞において追加して登録する情報

動詞、形容詞との共起情報(意味素性)

データベースのデータ項目との対応

対話管理部:

対話管理ルールベースに記述された規則に基づき、ユーザとの間の対話の履歴を対話履歴ベース上に保存・管理し、またユーザに対するシステムの応答内容を深層意味表現の形で生成する。第4章で詳しく述べる。

文音声合成部:

対話管理部において生成された応答内容の深層意味表現をもとに、ユーザに提示する音声を合成する。その処理手順としては、まず深層意味表現をもとに文法ルールベース上の規則に基づき焦点などの情報も考慮して統語情報の作成や述語の活用を行ない表層文を生成する。次にこの表層文や統語情報などをもとに、音韻ルールベース上の規則に従い、異音化処理を施したりイントネーション・アクセントなどの韻律的情報を生成することによって、単音・韻律記号列を生成する。そして最後にこの単音・韻律記号列をもとに音響処理部において音声合成器を駆動し、合成音声を生成しユーザに提示する。

3. 深層意味表現

本章では深層意味表現について、その具体的な定義を行なうこととする。

深層意味表現とは、表層文において表現されている意味内容に関する情報だけを抽出して表現したもので、統語情報などは含まれない。深層意味表現には一階述語論理や R.C.ShankによるCD表現などいくつかの表現方法が提案されているが、本システムにおいては以下の理由から格構造の概念を用いることとした。

・システムの目的上、対象となる言語は日本語だけであるので、中間言語表現を用いた機械翻訳システムなどのように言語間の差異を吸収するために深層意味表現としてかなり抽象的なレ

ベルのものを採用する必要がない。

・本システムへの入力として深層意味表現を用いるので、人間が実際に入力したい表層文から深層意味表現を容易に導出できる必要がある。

具体的な格構造としては、次のように記述することとした。

(述語 (意味付加) (文脈情報)

((深層格 格のfiller)

(深層格 格のfiller) ...)

この表現のうち、意味付加とは述語の活用と助詞、助動詞の付加によって表現できる意味情報を表わし、本システムではとりあえず表2に示すものを用意した。また文脈情報とは複数の格構造間の意味的接続や対話状況におけるこの格構造の果たす役割などに関する記述である(理由、前提条件の設定など)。さらに対話状況から焦点がおかれたり、表層表現において省略・照応表現を用いるべき深層格にはマークを付与する。

表2 本システムで用いる意味付与

否定 (Negative)	疑問 (Question)
丁寧 (Polite)	完了 (Perfect)
希望 (Desire)	

この格構造において用いる深層格としては、一般性を持たせるために機械翻訳システムの開発を目指したMuプロジェクトにおいて用いられている深層格の種類を参考にした⁽⁴⁾。Muプロジェクトにおいては動詞・名詞間の深層格として全部で34個が用意されているが、本システムにおいては、このうち対象となる分野の対話を表現するのに必要であると考えられる14個を用いることとした。本システムにおいて用いる深層格の一覧を表3に示す。

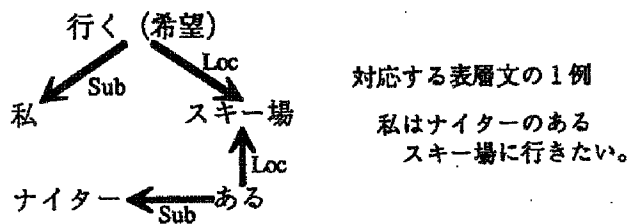
最後に例として、深層意味表現とこれを意味ネットワーク形式で表現したもの、及びこれによって表現される表層文の1例を図2に示す。

表 3 本システムにおいて用いる深層格

主体 (Subject)	時・終点 (Time-To)	場所・経過 (Space-Through)
対象 (Object)	時間 (Duration)	原因・理由 (Cause)
相手 1 (Partner)	場所 (Space)	手段・道具 (Tool)
時 (Time)	場所・始点 (Space-From)	目的 (Purpose)
時・始点 (Time-From)	場所・終点 (Space-To)	

(行く (希望) (φ)
 ((主体 私) (場所 スキー場[1])))
 (ある (φ) (限定)
 ((主体 ナイター) (場所 スキー場[1])))
 [1]は同じものを指していることを示す添字

リスト形式による格構造表現



ネットワーク形式による表現

図 2 深層意味表現の例

4. 対話管理部

対話管理部ではユーザとシステムの間の対話履歴の管理と、この履歴情報をもとにユーザからの入力に対してシステムが応答する内容を深層意味表現の形で生成する応答内容立案が行なわれる。

対話履歴の管理において行なうべき処理としては以下のものが挙げられる。

- 対話の流れの管理
- 省略・照応表現の処理
- 主題の管理
- 焦点の管理

また応答内容立案部において行なうべき処理としては以下のものがある。

- データベース検索
- 検索結果に基づく応答内容立案

これらの処理を行なうためには様々な形で規則(知識)を導入する必要があるが、話題に依存しない一般的な対話規則だけで十分な対話管理を行なうことが不可能なことは明らかで、話題などに依存した規則も必要となる。しかし本システムを構成するに当たっては、使用する知

識ベースが変更されてもシステム構成やアルゴリズムが有効であることを示すことを1つの目的としている。そこで、対話管理に関する規則を話題に依存する規則と話題に依存せず質問応答型対話一般について成り立つ規則とに分け、知識ベースを変えることによる変更部分をなるべく少なくかつ明確にするように配慮している。

具体的には、対話管理に関する規則を以下の3つに分類し、データベース内容の変更に伴う話題の変更やデータベース検索というタスクの変更(例えばエキスパートシステムとの対話というタスクへの変更)による対話管理規則の更新部分を明確化かつ局在化させている。

・質問応答型対話一般に関する規則

対話履歴の管理に関する規則など質問応答型対話一般において成り立つ規則

・話題に依存する規則

話題に固有の対話の進め方のプランを対話の状況ごとに記述した規則

・タスクに依存する規則

データベース検索というタスクに固有の対話の進め方のプランを記述した規則

本章では、対話履歴管理と応答内容立案の2つの処理を中心に説明を進め、その過程で3種類の規則に関して言及する。なお、対話管理部周辺の詳細図を図3に示す。

4-1. 対話履歴管理

対話履歴の管理の中で、最も重要なのは、ユーザとシステムの間で行われる対話の流れ(進行)を適切な形式で表現し、保存・管理することである。次節で説明する発話内容立案で、この保存内容が大きく関係してくるからである。本節ではこれらの処理を実現するための前提として、音声対話の特徴に関する考察結果⁽⁵⁾に基づく対話の流れの表現方法について説明し、さらにそれに基づく対話の流れの管理、および



このような対話の流れを表現する方法として、本システムでは対話の基本ルーチンに基づいたものを用いている。すなわち、図4の①の主題の設定及び質問の発話によって基本ルーチンが

履歴管理部は新たに基本ルーチンを生成するとともに、質問内容を応答内容立案部に送る。その結果として質問に対する回答が得られた場合はその深層意味表現を出力して基本ルーチン

を終了し、1つ外側の基本ルーチンに制御対象を戻す。一方、回答が得られない場合は次節で述べるようにその状況に応じて話題依存・タスク依存の規則を適応してユーザに対して質問や要求などの協調的割り込みの発話を行なうので、それに対応する新たな基本ルーチンを生成する。

② 要求系のユーザ入力時

基本ルーチンの管理という点では、①の質問系の処理の場合と似ており、まず要求系の入力によって基本ルーチンを生成、次にその内容を応答内容立案部で処理した結果、要求に対する確認、受諾、要求内容の実行が得られた場合には、応答内容の深層意味表現を出力するとともに、基本ルーチンを終了し1つ外側の基本ルーチンに制御対象が戻る。また得られなかった場合には、話題依存、タスク依存の規則を用いてユーザに対して協調的割り込みの発話を行なう。

③ 応答系のユーザ入力時

この入力が許容されるのは、いちばん内側の（現在処理対象の）基本ルーチンがシステムの発話によって生成され、しかも終了していないときであり、その場合以外は不適切入力としてその旨をユーザに伝達する。条件を満たす場合はその内容がこのルーチンの終了条件を満たすか（システムが要求した情報と深層格、意味素性などが一致するか）を判断し、適切であればこの基本ルーチンを終了させ、1つ外側の基本ルーチンに制御対象が戻る。また回答として不適切な場合はやはり不適切発話として扱う。

④ 外側の基本ルーチンに制御対象が戻ったときの処理

①から③の処理のなかで基本ルーチンが終了し1つ外側の基本ルーチンに戻ったときの処理は、次の2つの場合に分けて行われる。

・戻ってきた基本ルーチンがユーザの発話によって生成されたとき

基本ルーチンが生成されたときのユーザ発話の内容に従って、ユーザに回答が提示できるまで①（または②）の処理を行なう。

・システムの発話によって生成されたとき

システムはユーザからの入力待ちの状態になるが、しばらく待って入力がない場合は、システムからユーザに対して入力を催促する発話を行なう。

⑤ その他

ユーザの発話中に未登録語がある場合に、それについて質問するなど、人間同士の対話にお

いて一般に行われる言語行動をプランの形で規則として記述しておき、必要に応じて適用する。また次項で述べる省略・照応表現に関する規則も質問応答型対話一般に成り立つという点で一般規則に含まれる。

以上の規則に従って基本ルーチンの生成、終了を行なうことによって個々の発話の対応関係を記述できる。そして各発話に対する深層意味表現とこの基本ルーチンを一緒に対話履歴ベースに保存することによって、対話の流れを管理することができる。

しかし以上は協調的割り込みに対応した処理であって、競合的割り込みについてはまだ十分に考慮していない。今後の検討課題として残されている。

またこのように基本ルーチンを用いて対話の流れを表現することによって、システムは基本ルーチンによる発話の対応関係から次のユーザからの入力の内容を対話状況に応じてある程度類推することも可能となり、これは将来的には音声理解を行なう際の有用な情報の1つとして利用し得ると考えられる。

4-1-3. 省略・照応表現処理

省略・照応表現の処理には、ユーザ入力における省略・照応表現に対する補完と、システム発話の際にどの部分に省略・照応表現を用いるかという2つの側面がある。そしてこれらの処理を行なうために必要な情報として、システムは対話履歴情報の一部として表3に示された各深層格ごとにこれまでの対話の中でその格を占めた最新の単語と、その単語が使われた発話を対話履歴ベースの中に記録しておく。

ユーザ入力の補完処理が行われる場合としては、次の2つの状況を想定している。

・ある深層格に照応表現（代名詞）が用いられているとき

・ある述語の必須格が省略されているとき

このような場合には、その対話履歴ベース中のその深層格の保存値でその部分の補完を行なう。

またシステム発話においては、応答内容立案部によって生成された応答内容を記述する深層意味表現において、ある深層格を占める単語が対話履歴ベース中の保存値と同じであり、かつその単語が使われてから現在までの発話数が一定値以下である場合に、その格が必須格の場合

ユーザ要求情報	スキー場までの所用時間	スキー場の紹介
検索必須情報	対象スキー場名 利用交通手段	いつ行くか 誰と行くか 滞在期間
検索任意情報	φ	ユーザが任意に指定

図7 話題依存規則のひな型の例

は省略表現を、任意格の場合は照応表現を用いる。

4-2. 応答内容立案部

履歴管理部にて処理されたユーザ入力と、これまでのユーザとシステムとの間の対話によって得られている情報に基づいて、応答内容立案部では必要に応じてデータベース検索を行ない、ユーザに対する応答内容を深層意味表現の形で生成する。そのため必然的に前節で述べた履歴管理部に比べて対話の話題やデータベース検索というタスクに依存した形での処理の比重が大きくなっていく。

本節ではまず応答内容の立案に必要な話題やタスクに依存した規則について言及した後、どの様にデータベースの検索を行ない、応答内容の意味表現を生成するかについて説明する。

4-2-1. 話題依存規則

話題依存規則はユーザとシステムの間で行われる対話の話題、すなわちデータベースによって提供される情報の内容に依存する規則である。具体的には、ユーザの質問内容が漠然としていてそれだけの情報ではデータベース検索ができず、ユーザの質問・要求に回答できない場合（例：「今度スキーに行きたいのですが、どこがいいですか」）に、どの様に対話を進めて行くかを対話状況に応じた形で記述しておく。

本システムでは、対話の話題としてスキー場の案内及び紹介を想定しているので、あらかじめ対話の状況によってデータベース検索を行なうのに必要な最小限の情報項目を記述したひな型を用意しておき、ユーザの発話からこの項目を満たす情報が得られきれないときは、その情報をユーザに質問することによって求め、それ

からデータベース検索を行なう。このようなひな型の例を図7に示す。そしてこのようなひな型を追加・拡充していくことによって、システムをユーザとより自然に対話ができるように改良していくことが可能となる。

またこのひな型の選択は、図6のユーザ入力の分類結果や発話中の述語、ユーザ要求情報の意味素性などの情報に基づいて行なわれる。

4-2-2. タスク依存規則

タスク依存規則はこのシステムがユーザとの対話を通じてどの様なタスクを実行するかという点に依存する規則である。本システムの場合はデータベース検索を行なうことを想定しているので、タスク依存規則は図8のように記述することができる。

- ・ 検索条件に該当するものが多すぎる
該当件数を提示
- ・ 追加条件要求または全部提示するかの確認
- ・ 質問内容がデータベースの項目にない
その項目に関する情報がないことを伝達

図8 タスク依存規則（一部）

4-2-3. データベース検索とシステム発話内容の決定

実際のデータベース検索処理はデータベースの構造に依存してくるので、データベースとのインターフェースもデータベースに依存して変更を加える必要がある。

その具体的な方略としては、話題依存の規則によってデータベース検索に必要な十分な情報がユーザから得られた段階で、単語辞書中のデータベースとの対応に関する記述を参照して、名詞的概念から検索するデータベースのデータ項

目を決定し、述語的概念からその検索条件を決定する。

システムの応答内容の意味表現の生成過程は、3種類の規則のどのレベルで応答が生成されるかによって変わるが、例として話題依存規則によって応答内容が生成される場合を説明する。

この場合は応答内容は話題依存規則のひな型を基に生成され、ひな型に記述された検索必須情報が全て求められていない場合は、その部分を疑問詞にして深層意味表現中に埋め込む。また質問に対する応答を行なう場合は、データベース検索の結果得られる内容を深層意味表現中に埋め込む。この時の深層意味表現の述語の選択は、あらかじめひな型ごとに分類しておき、それに従って行なう。

5. おわりに

筆者らが現在進めている日本語による音声応答システムの実現を目指した研究について、システム全体の概要及び対話管理部における処理について概説した。システムを構成する過程において、人間の音声による対話活動に関していくつかの知見および仮説を組み入れた。

現在、システムのプロトタイプを計算機上に実装する作業を進めている。将来的には音声対話に固有の音声現象、特に韻律的特徴に関する分析を進め、それらの知見をシステムに組み込むことによって、よりUser friendlyなシステムとして発展させる。

謝辞

本研究に関し、日頃ご指導いただく東京理科大学・藤崎博也教授に、謝意を表する。

参考文献

- (1) Young, S.J and Fallside, F: "Speech Synthesis from Concept: A Method for Speech Output from Information Systems," J. Acoust. Soc. 66(3), pp.685-695 (1979).
- (2) Litman, D.J and Allen, J.F: "A Plan Recognition Model for Subdialogues in Conversations," Cognitive Science, Vol.11, pp.163-200 (1987).
- (3) 飯田仁, 有田英一: "4階層プラン認識モデルを使った対話の理解," 情報処理学会論文誌, Vol.31 No.6, pp.810-821 (1990).

- (4) 辻井潤一: "機械翻訳システムにおける辞書の構成," 野村浩輝編 "言語処理と機械翻訳", pp.71-105 (1991).
- (5) 浅野康治, 広瀬啓吉, 藤崎博也: "音声応答システムにおける対話管理手法," 日本音響学会秋季講演論文集, 3-P-17, pp.325-326 (1991).
- (6) 飯田仁: "異言語間対話を目指す端末間通訳モデル," 認知科学の発展, Vol.1, pp.112-140 (1988).

●浅野康治 広瀬啓吉(東京大学 工学部)
藤崎博也(東京理科大学 基礎工学部)

1. はじめに

筆者らは、従来より知識表現からの音声合成に関する研究を進めており、その具体例としてシステムが合成音声で応答する質問応答システムをとりあげている⁽¹⁾。この質問応答システムにおいては、ユーザの入力からユーザの望む情報を適切に選択し、伝達するという円滑なコミュニケーションを実現するために、ユーザとシステムの間での対話をシステムが十分に管理・把握する必要がある⁽²⁾。本報では実際に人間が音声によって行った対話(主として一方があることに関する質問を、他方がそれに対する応答を行う質問応答タスク)をテキスト化したものを分析した結果として、まず音声対話に特徴的な点について考察した後、それらに対処するための対話管理の方策及び対話管理に必要不可欠な対話履歴ベースにどのような情報を蓄積する必要があるかを検討した結果について報告する。

2. 音声対話の特徴とそれへの対処

音声対話の管理を考えるにあたって、音声対話がキーボード間対話(例えばUNIXのtalk)などテキストによる対話とどのような点で異なり、それに対してどう対処するかという方針を立てる観点から、質問応答タスクでの音声対話の分析を行った。音声対話に固有の特徴としては以下のものが考えられる。

- ①「あー」、「えー」などの直接対話の流れに関係しない発話や、言いよどみ、言い直しが頻繁に生起する。
- ②対話の途中に、相づちや質問などが割り込み的に発生する。そして、それによって一時的に質問と応答の立場が容易に入れかわりうる。
- ③聞き取りミスによる誤解等が生じやすい。
- ④対話内容の保持が人間の脳によるため(テキストとして残らない)、過去の内容との重複や、再確認などが生じる。

これらのうち、①については、人間は発話内容を考えながら発声するためにこのような現象が生じる。そのためシステムが音声で応答する場合には、この現象を実現する必要はないと考えられる。これに対し②～④については、対話の流れに直接関係する問題で、音

声応答を目的とする対話管理においてはこれらに対処する必要がある。このうち、②については対話管理部において現在の対話の状態を把握することによって対処する。また③、④については過去の対話履歴を蓄積し、その内容をもとに現在の対話状態を監視することによって対処できると考えられる。

3. 対話管理部における対話の流れの表現

3-1. 対話の流れ

質問応答タスクにおける対話は、通常図1に示す3つの発話が1組となって1つのルーチンを形成し、それらの繰り返りで全体の対話が構成されていると考えることができる。

①質問する話題の設定

②質問

③質問に対する応答

図1 対話の基本ルーチン

このうち、①と②は1つの発話で一緒に行われる場合も多い。

音声による対話ではテキストによるやりとりよりもはるかに容易に相手の発話に対して割り込めること、音声対話における発話では1つの文で必要な情報すべてを表現するのは希であることなどから、図1の対話の流れに対する割り込みが数多く生じる。この割り込みは協調的なものと競合的なものの2種類に分類することができる⁽³⁾。

協調的な割り込みとは、相手の発話内容の一部が理解できずに質問する場合など、現在の対話の流れに沿った上での割り込みで、この場合は割り込みで要求された処理を実行した後に、対話の流れをもとの状態に戻すことができる。また、この協調的割り込みが生起する割り込みの大半を占める。

一方、競合的な割り込みとは、一方の発話目的が達成されないうちに、他方から現在の対話の流れに無関係な別の話題に関して発話が行われるというものである。

3-2. 対話管理部での対話の流れの表現

対話管理部でこれらの対話の流れを把握する方法として、図1に示した対話の基本ルーチンをもとに対話の流れを表現することが考

*Conversation Management Method for Speech Response System

By Yasuharu Asano, Keikichi Hirose (Faculty of Engineering, Univ. of Tokyo)

and Hiroya Fujisaki (Faculty of Fundamental Engineering, Science Univ. of Tokyo).

◎浅野康治 広瀬啓吉 (東京大学 工学部)
藤崎博也 (東京理科大学 基礎工学部)

1. はじめに

筆者らは、従来よりユーザの求める情報を連続音声の形で提示する音声応答システムに関する研究を進めている。そして、その中でもシステムとユーザとの間で行なわれる対話を管理して両者のコミュニケーションを円滑に進めるための対話管理部の重要性を指摘し、それに要求される機能についての検討を進めてきた⁽¹⁾。本報では、この対話管理部において行なうべき処理についてこれまでの研究の概略をまとめた後、対話管理における機能の1つである対話における主題の管理と焦点の管理について、その具体的な手法について報告する。

2. 対話管理部の概要

これまで行なってきた検討結果に基づき、具体的に対話管理部においてどの様に処理が進められていくかについて概説する。

対話管理部において行なわれる処理は大きく次の2つに分けることができる。

- ・対話履歴管理
- ・応答内容立案

対話履歴の管理においては、対話の基本ルーチン⁽¹⁾に基づく対話の流れの管理を行なうとともに、省略・照応表現の処理や後述する対話における主題の管理、焦点の管理を行なう。一方、応答内容の立案においては、ユーザからの要求内容に応じて応答内容の意味表現の作成を行ない、その際に必要に応じて知識ベースにアクセスし必要な情報を得る。

以上のような処理を行なうにあたっては、そのための規則が必要となるが、汎用的な規則だけでは円滑な対話は不可能で、対話の話題や対話によって遂行するタスクに依存した規則が必要不可欠である。そこで本システムでは使用する知識ベースが変更されてもそれにともなう規則の変更箇所を明確かつ局在化することによって、応答システム自体の構成の汎用性を高める

ように配慮している。具体的には、対話管理に関する規則を ①質問応答型対話一般に対して成り立つ規則、②対話の話題に依存する規則、③対話によって遂行されるタスクに依存する規則 の3種類に分類している⁽²⁾。

以上をまとめると対話管理部の構成は図1

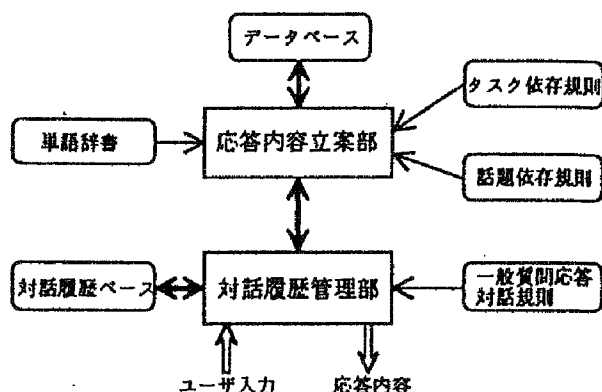


図1 対話管理部の構成

U: ○○スキー場へは何時間かかりますか。

(設定、質問)

S: 車で行きますか、JRですか。

(設定、質問)

U: 車で行きます。 (応答)

S: それでは4時間くらいです。 (応答)

(1) ユーザ (U) によるルーチンの生成

(2) システム (S) によるルーチンの生成

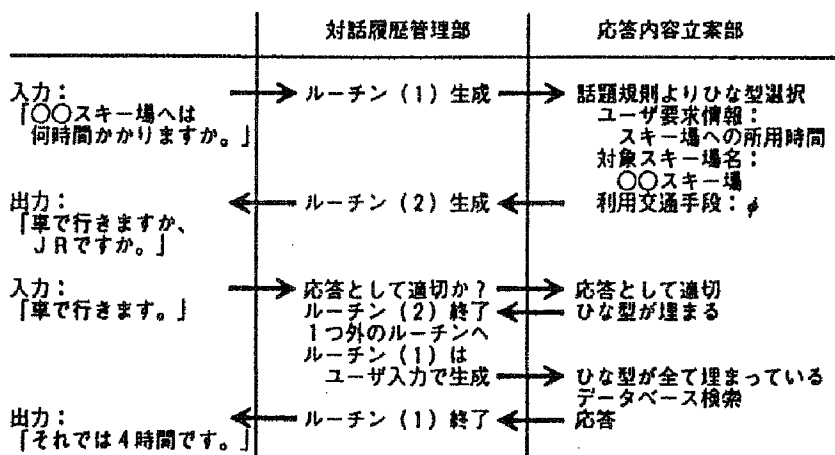


図2 対話管理部の動作例

*Realization of Dialogue Management Part for a Speech Response System

By Yasuharu Asano, Keikichi Hirose (Faculty of Engineering, Univ. of Tokyo)
and Hiroya Fujisaki (Science Univ. of Tokyo).

のようになり、これによる対話管理部の動作の具体例を図2に示す。

3. 主題の管理

一連の対話が行なわれる過程で、ある時点での対話の内容(主題)は時間の進行につれて前後に関連を持ちつつ変化していく。この変化の過程を対話管理部で管理することによって、ユーザに対する応答をよりの確に生成できるとともに、将来的に本応答システムへの入力を音声を通して行なう場合に、その処理部へ有用な情報を提供できると考えられる。

なお、本報においては「話題」と「主題」を以下のように使い分けている。「話題」とはシステムで扱う対話の題材を指し、本システムの場合はスキー場に関する紹介・案内である。一方「主題」は対話の部分部分で変化するより小さいテーマを指すものとする。

主題の管理は、機能的には対話履歴の管理の一部をなすものと考えられるが、対話の内容に関係してくるので、対話の話題に依存した規則なしに行なうことは不可能である。そこで、本システムでは主題の管理を以下のように行なうこととした。

前記の対話管理に関する規則のうち、②の話題依存規則には、ユーザ要求情報とそれを得るために必要となる検索必須情報と検索任意情報とが図3に示すようなひな型の形で記述されており、これによって対話の進め方の概略が決まる。すなわち、主題の管理は、複数用意されているひな型をどの様を選択して使用して行くかを管理することによって行なうことができる。

この管理にあたっては、ひな型中のユーザ要求情報、検索必須情報、検索任意情報を施設、交通、時間などより抽象的な項目ごとにグループ化を行ない、各グループ間の対応関係を記述することによってひな型の間の関連付けを行なう。またグループごとに関連する語彙やデータベースのデータ項目をまとめ、これによってシステム発話の際の語彙の選択や、逆にユーザが使用した語彙から使用するひな型の選択を容易にすることができる。

	ひな型1	ひな型2
ユーザ要求情報	スキー場への所用時間	スキー場の紹介
検索必須情報	対象スキー場名 利用交通手段	いつ行くか 誰と行くか 滞在期間
検索任意情報	φ	ユーザが任意に指定

図3 話題依存規則のひな型の例

4. 焦点の管理

焦点とは、対話のある時点(ある対話の状況下)において、1つの発話によって相手に伝達される情報の中心となるもの、すなわち発話者が相手に最も把握してもらいたいと考えている情報であり、この管理は音声応答システムにおける対話管理においては以下の2つの面から必要不可欠な機能である。

まずシステムの次の発話の焦点の位置を制御することによって、音声の合成時に焦点がある部分を強調し(ピッチをあげる、発話速度を遅くする)、ユーザに対してより理解しやすい合成音を生成することが可能となる。具体的にどの情報に焦点をおくかは、人間が行なった質問応答の対話例を参考にして、図4に示すように規則化した。

- ・ユーザの質問に対する回答の情報

例:「ナイターができるのは〇〇スキー場ですか。」

「いいえ、××スキー場です。」

- ・システムがユーザに対して必要な情報を問い合わせる場合、その必要となる情報

例:「〇〇スキー場へは何時間かかりますか。」

「何で行きますか。」

(例文の下線部に焦点を設定)

図4 焦点の管理に関する規則

また、省略・照応表現は前の発話で焦点のある部分に対して使用される場合が多いので、焦点管理によって省略・照応表現によって欠落した情報の補完が容易になる。

5. おわりに

音声応答システムにおける対話管理に関して、これまでの検討結果から得られた知見や仮説をまとめ、それらを反映した形での対話管理部の実現による処理の流れと、新たに検討を行なった主題の管理と焦点の管理について述べた。

今後は現在進めている対話管理システムの計算機上への実装に、今回検討を加えた点も組み入れ、システムの有効性について検討を加える予定である。

参考文献

- (1) 浅野他:「音声応答システムにおける対話管理手法」音講論, 3-P-17 (1991-10).
- (2) 浅野他:「相互主導型対話機能を持つ音声応答システムとその対話管理手法」信学技報, SP91-110 (1992-1).

J. Acoust. Soc. Jpn. (E) 11, 4 (1990)

Half-syllabic units for speech processing—An automatic segmentation

Keywords: Units for speech processing, Half-syllabic units, Segmentation, Dynamic spectral feature

PACS number: 43.72.Ar

M. Nakatsui* and R. Tanaka**

*Communications Research Laboratory,
4-2-1, Nukukita-machi, Koganei, 184 Japan

**National Space Development Agency,
2-1-1, Sengen, Tsukuba, 305 Japan

(Received 24 August 1989)

1. Introduction

The half-syllabic units proposed here are units each of which has segment boundaries at steady portions and preserves a transition between two phonetic units.¹⁾ Segment boundaries are basically determined by the minima (valleys) of gross spectral variation measure. The spectral variation measure is defined as the root-mean-square value of the slopes of the weighted regression lines calculated from LPC cepstrum parameters over several frames. The maxima (peaks) of the measure will serve as the reference points for further processing.

In speech synthesis by rule, it is primarily important to select synthetic units that have reasonably small size of inventory to represent spoken utterances and, at the same time, are easily concatenated. In speech analysis-system at very low-bit-rates such as phonetic vocoding, the units must, further, be automatically segmented and be suitable for interpreting into or matching with the reference units. These requirements on segmentation and matching or labelling are expected to be satisfied for speech recognition system in many cases and for providing useful tools for automatic generation of the inventory of concatenative units.

Syllables and Half-Syllables

One of the selections for the unit to be used in concatenation-based speech processing is the syllable. There have been several discussions and experiments on syllable as recognition unit.²⁻⁵⁾ The syllable has been also used as a unit in synthesis by rule of Japanese.⁶⁾ One of the disadvantages to using syllables as units is that the size of inventory representing spoken utterance is large. This problem can be solved by introducing smaller units such as the half-syllabic units proposed here, since much of the co-articulation among phonetic units is associated with transition regions and since

boundaries at the steady portions outside transitions are easily definable. There exist similar units known as dyads,⁷⁾ diphones,⁸⁾ or demisyllables⁹⁾ which have the common concept of incorporating the transition between phonemes. The context-dependent diphones have been utilized in constructing a phonetic vocoding system.¹⁰⁾ The demisyllables originally proposed for use in a high-quality concatenative speech synthesis⁹⁾ have been successfully applied to constructing concatenative templates in the word recognition for large vocabularies.¹¹⁾ Japanese half-syllabic units have been used in speech synthesis by rule¹²⁾ and word recognition based on Hidden Markov Models.¹³⁾

Dynamic Spectral Feature

The gross spectral variation measure derived from a series of LPC cepstrum coefficients has been proposed as a dynamic measure investigating individuality of utterances.¹⁴⁾ This dynamic measure has been used in the study on Japanese CV-syllable perception and it has been shown that dynamic spectral feature plays a primary role in phoneme perception.¹⁵⁾ Usefulness of the dynamic measure in comparison with its static counterpart has also been shown in word recognition experiment.¹⁶⁾ The dynamic measure has also been applied to the segmentation in a very low-rate speech coding where boundaries of the pattern are defined by the maxima of the measure.¹⁷⁾

Our expectation for the units is in the relatively small size of inventory in representing Japanese utterances, since Japanese has relatively simpler syllable organization than that of English. Our ultimate objectives is to provide nearly universal units suitable for processing spoken Japanese. As the first step to that goal, our current interest is in confirming whether the proposed units meet the basic requirements that they would be

- 1) automatically and reliably segmented,
- 2) closely related to certain linguistic units, and
- 3) suitable to acoustic phonetic observations

in the course of constructing the analysis-synthesis system like segment vocoder. This paper reports a preliminary experiment on segmentation of speech signal into the units proposed and some observations of the result with respect to the above requirements.

2. Segmentation Algorithm

Speech sample is bandlimited to 4 kHz and digitized to 12 bits at sampling frequency of 10 kHz. Linear prediction analysis is carried out on a frame-by-frame basis (100 frames/s). Additional acoustic parameters currently used are a log power P , a zero-crossing count Z , a count for sign change of waveform X , and the first order PARCOR coefficient k_1 . The spectral

variation measure $D(j)$ for j -th frame is calculated by

$$D(j) = \left[\frac{1}{12} \sum_{i=0}^{12} \{u(i)a(i,j)\}^2 \right]^{1/2}, \quad (1)$$

where weight $u(i)$ is currently one for all i and $a(i,j)$ is the i -th coefficient of the weighted regression line of LPC cepstrum parameter over several frames. A triangular weighting function is currently applied over seven frames.

With these acoustic parameters, signal processings on input speech are basically carried out in the following steps (descriptions in parentheses are associated with indications in Fig. 1):

- 1) appointing candidates for segment boundaries at local minima of spectral variation measure (vertical lines),
- 2) adjusting the segment boundaries by start and end points of speech interval (S and E),
- 3) classifying the boundaries into sub-groups of phonetic units and assigning candidates of vowel identity,
- 4) assigning the reference points at maxima of the variation measure (dotted vertical line) for time alignment in spectral matching with the reference patterns, and
- 5) adopting weights for pattern matching inversely proportional to the normalized values of the spectral variation measure.

Among those steps, 3) to 5) are beyond the scope of this report. However some preliminary trials will be shown later. As for 2), a hysteresis characteristic is given to the decisions of speech interval (from S to E) providing two levels of thresholds for the log power P and the decisions for the nonspeech interval associated with intervocalic unvoiced stops are stabilized by referring the count of sign change X . The minima (valleys) just before S and that just after E were assigned as boundaries of the utterance.

3. Result of Preliminary Experiment

Sixty names of Japanese cities spoken by a male adult were used as the test material for segmentation process. It was estimated that the test material consisted of 455 half-syllabic units by our visual inspections.

Segmentation

Figure 1 shows an example of segmentation where the segment boundaries are denoted by vertical lines and reference points for matching are denoted by the dotted vertical lines. Result of an automatic segmentation of the test material is summarized in Table 1. Correct rate of segmentation is more than 94%. Most of the deletions of segment boundaries at word-middle are associated with intervocalic [r] and [g] sounds. These problems are going to be solved by the test material having wider spectral bandwidth. It is revealed that problems concerning deletions at word-initial and insertions at word-final are also due to inadequacy of the test materials such as low signal-to-noise ratio and

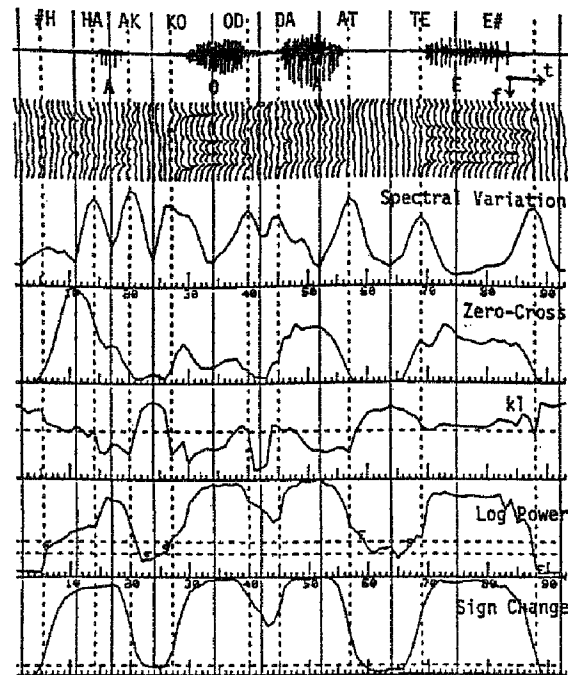


Fig. 1 An example of segmentation and acoustic parameters.

Table 1 Segmentation errors for 455 segments.

Position	Initial	Middle	Final	Total
Deletion	7	7	0	14
Insertion	0	11	1	12
Total	7	18	1	26

over-cuts at the beginning and the end of utterances. So, new test material suitable for our experiment is under preparation, because the current sample has been prepared for other experimental purpose.

Most of insertions of segment boundaries, extra boundaries than expected, are associated with nasal and unvoiced stop consonants. It is observed that extra segments correspond to nasalized vowels and aspirations after stop bursts. The detailed observation for much speech material from the point of view of acoustic phonetics should be made in order to give such solution and interpretation systematically.

Some Observation on Segments

Signal processings described below have not been fully automatized yet and, further, most of the observations have been based on a small set of test material. Alphabets at the top of Fig. 1 are our tentative labelling for the segments (units). Segment boundaries are first classified as either vowels or one of a consonantal groups such as voiced-stops and unvoiced-fricatives using a set of acoustic parameters. Spectral distances

between spectral frame of the boundary and single frame reference patterns including isolated five vowels and nasal murmurs were used as additional information in the classification. Alphabets on the segment boundaries just below waveforms in Fig. 1 denote the first candidates of vowel identity showing minimum spectral distance. Ninety percent of vowel boundaries are identified as the first candidates and the remaining ten percent as the second for a sub-set of the test material having 40 vowels.

Linear spectral matchings of the CV-type segments with the CV-syllable reference patterns prepared from CV-syllable utterances were tried using the classification results and the first and second vowel candidates described above. In the matching, time alignment between the segment and the reference pattern was adjusted in such a way that the reference points of both patterns coincide. It is observed that correct CV-syllable appears within the top three candidates for most cases in this arrangement.

4. Concluding Remarks

Although our experimental evidence is at quite a primitive stage, the half-syllabic units proposed seem to have potential to meet three basic requirements described above. Among many problems left to be solved, our current interests are in (1) preparation of speech material suitable for our objectives, including city-names at different speeds of utterance and conversational utterances, (2) improvement and tuning of the segmentation algorithm applicable for these speech data, and (3) the detailed observation of the units in acoustic phonetic aspect and systematic organization of classification algorithm. Another problem of the units as synthetic ones is in that special care should be taken when consonantal segment boundaries are concatenated.

References

- 1) M. Nakatsui and R. Tanaka, "Half-syllabic units segmented with spectral variation," Proc. Autumn Meet. Acoust. Soc. Jpn. 1-4-9, 37-38 (1985) (in Japanese).
- 2) O. Fujimura, "Syllable as a unit of speech recognition," IEEE Trans. Acoust. Speech Signal Process. ASSP-23, 82-87 (1975).
- 3) P. Mermelstein, "A phonetic-context controlled strategy for segmentation and phonetic labelling of speech," IEEE Trans. Acoust. Speech Signal Process. ASSP-23, 79-82 (1975).
- 4) M. J. Hunt, M. Lenning, and P. Mermelstein, "Experiments in syllable-based recognition of continuous speech," Proc. ICASSP '80, Denver, 880-883 (1980).
- 5) H. Fujisaki, K. Hirose, T. Inoue, and Y. Sato, "Automatic recognition of spoken words from a large vocabulary using syllable templates," Proc. ICASSP '82, Sandiego, Vol. 3, #26.12 (1982).
- 6) Y. Tohkura and Y. Sagisaka, "Synthesis by rule using CV-syllable and its speech quality," Tech. Rep. Speech, Acoust. Soc. Jpn. S80-47 (1980) (in Japanese).
- 7) G. E. Peterson, W. S.-Y. Wang, and E. Sivertsen, "Segmentation techniques in speech synthesis," J. Acoust. Soc. Am. 30, 739-742 (1953).
- 8) N. R. Dixon and H. D. Maxey, "Terminal analog synthesis of speech using the diphone method of segment assembly," IEEE Trans. Audio Electroacoust. AU-16, 40-50 (1968).
- 9) O. Fujimura, M. J. Macchi, and J. B. Lovins, "Demisyllables and affixes for speech synthesis," Proc. 9th ICA, Madrid, #I-107, 513-514 (1977).
- 10) R. Schwartz, J. Klovstad, J. Makhoul, and J. Sorensen, "A preliminary design of a phonetic vocoder based on a diphone model," Proc. ICASSP '80, Denver, 32-35 (1980).
- 11) A. E. Rosenberg, L. R. Rabiner, J. G. Wilpon, and D. Kahn, "Demisyllable-based isolated word recognition system," IEEE Trans. Acoust. Speech Signal Process. ASSP-31, 713-726 (1983).
- 12) K. Fushikida and Y. Mitome, "Japanese speech synthesis by rule using Formant-CV, VC compilation method," National Conf. Rec., 1981 Inf. Syst., Inst. Electron. Commun. Eng., #104, 1-101 (1981) (in Japanese).
- 13) K. Yoshida, T. Watanabe, and S. Koga, "A speech recognition method based on demi-syllable Hidden Markov Models," Autumn Meet. Acoust. Soc. Jpn. 2-P-24, 249-250 (1988) (in Japanese).
- 14) S. Sagayama and F. Itakura, "On individuality in a dynamic measure of speech," Spring Meet. Acoust. Soc. Jpn. 3-2-7, 589-590 (1979) (in Japanese).
- 15) S. Furui and M. Akagi, "On the role of spectral transition in phoneme perception and its modeling," Proc. 12th ICA, Toronto, #A2-6 (1986).
- 16) S. Furui, "Speaker-independent word recognition using dynamic features," IEEE Trans. Acoust. Speech Signal Process. ASSP-34, 52-59 (1986).
- 17) Y. Shiraki and M. Honda, "Very-low-rate speech coding using time space spectrum patterns," Tech. Rep. Speech, Acoust. Soc. Jpn. S84-06 (1984) (in Japanese).



Mamoru Nakatsui received the B. Eng. degree in 1963 from the University of Electro-Communications and the Dr. Eng. degree in 1976 from the Tohoku University. In 1963 he joined the Radio Research Laboratory (now called Communications Res. Lab.) of MPT, where he is now a Director of Communications Technology Division. From 1978 to 1980 he was an invited Professor of INRS-Telecommunications, University of Quebec, Canada. His main interests lie in speech processing and artificial intelligence.



Ryoji Tanaka received the B. E. degree in 1966, and the M. E. degree in 1968, respectively, both from the University of Tokushima. From 1968, he worked as a Research Engineer in the Radio Research Laboratories, Ministry of Posts and Telecommunications. From 1977

to 1978, he was a study member of Laboratoire d'Informatique pour la Mécanique et les Sciences de l'Ingénieur-CNRS, France. From 1987, he is a Senior Engineer in the Tsukuba Space Center of the National Space Development Agency of Japan.

Language processing in connection with speech translation at ATR Interpreting Telephony Research Laboratories

Akira Kurematsu, Hitoshi Iida, Tuyoshi Morimoto

ATR Interpreting Telephony Research Laboratories, Sanpeidani, Inuidani, Seika-cho, Soraku-gun, Kyoto 619-02, Japan

and

Kiyohiro Shikano

NTT Human Interface Laboratories, Midori-cho, Musashino-shi, Tokyo 180, Japan

Received 1 August 1990

Abstract. The relationship between speech and language processing is an important problem to be solved in order to achieve continuous large-vocabulary speech recognition for a speech translation system or the human interface of a man-machine system. For the recognition of large-vocabulary continuous speech, first the phonemes are recognized by HMM (Hidden Markov Model). A generalized LR parser is introduced to predict next words/phonemes. The Japanese utterance is successfully recognized by the combined HMM-LR parser (HMM-LR). Many phrase candidates are filtered out of the speech recognition system through the use of linguistic information.

An experimental system which translates spoken Japanese into English (SL-TRANS) has been implemented. The translation method consists of analysis, transfer and generation processes. A new method incorporated in the system analyzes the expressions of linguistic intention meanings in utterances. In applying this translation method to a goal-oriented dialogue corpus, involving inquiries and explanations regarding an international conference, the experimental system shows advantages for translating Japanese dialogues into English.

Zusammenfassung. Die Beziehung zwischen der Verarbeitung gesprochener Sprache und Sprache in Textform ist ein wichtiges Problem, das bei Spracherkennungssystemen mit großen Wortschätzen (für ein Sprachübersetzungssystem oder ein akustisches Mensch-Maschine-Kommunikationssystem) zur Lösung ansteht. Zur Erkennung kontinuierlich gesprochener Sprache mit großen Wortschätzen werden die Phoneme zunächst mit Hilfe eines Hidden-Markov-Modells identifiziert. Hierbei wird ein verallgemeinerter LR-Parser verwendet, um das jeweils nächste Wort bzw. Phonem vorauszusagen. Japanische Sätze werden mit Erfolg durch das kombinierte System aus Hidden-Markov-Modell und LR-Parser erkannt. Durch die Verwendung linguistischer Information werden zahlreiche Satzhypothesen ausgefiltert.

Implementiert wurde ein experimentelles System, das gesprochenes Japanisch ins Englische übersetzt. Die Methode der Übersetzung besteht aus den Stufen Analyse, Transfer und Generierung. Eine neue Methode, die hier realisiert wird, analysiert die linguistische Absichtsaussage in den Sätzen. Als Anwendungsbeispiel wird dieses Übersetzungssystem auf ein zielorientiertes Dialogkorpus ("Fragen und Erklärungen in Verbindung mit einer internationalen Konferenz") angewendet. Das experimentelle System zeigt Vorteile bei der Übersetzung Japanischer Dialoge ins Englische.

Résumé. Il faut résoudre l'important problème de la relation entre le traitement de la parole et le traitement linguistique pour pouvoir utiliser la reconnaissance de la parole continue avec de grands vocabulaires dans des systèmes de traduction de la parole ou comme interface de système homme-machine. Dans une première phase, notre système reconnaît les phonèmes à l'aide de chaînes de Markov cachées (HMM). Ensuite, un analyseur grammatical LR généralisé est utilisé pour prédire les mots et phonèmes suivants. La combinaison des deux méthodes (HMM-LR) permet de reconnaître une phrase japonaise. L'utilisation d'informations linguistiques permet d'éliminer un grand nombre de phrases candidates.

Un système expérimental de traduction japonais-anglais a été implémenté. La traduction se déroule en trois phases: analyse, transfert, génération. Une nouvelle méthode incorporée dans le système analyse l'expression des intentions contenues dans les énoncés. Nous avons appliqué cette technique à un système de dialogue orienté se rapportant à des questions et à des explications concernant une conférence internationale. Ce système bien qu'expérimental semble bien adapté à la traduction vers l'anglais de dialogues en japonais.

Keywords. HMM (Hidden Markov Model), HMM-LR parser, spoken language translation, HPSG (Head-Driven Phrase Structure Grammar), Unification-based grammar.

1. Introduction

An automatic telephone interpreting system requires the development of various constituent technologies such as speech recognition, machine translation, and speech synthesis. These technologies are used in building a speech recognition system for spoken word comprehension, a speech translation system for spoken language conversion, and a speech synthesis system by which a computer will produce the voice of the "interpreter". These individual subsystems will then be integrated to form an automatic telephone interpretation system.

Integrating speech and language processing is an important problem to be solved in achieving continuous large-vocabulary speech recognition for a speech translation system or human interface of a man-machine system. Integrating speech and language processing is tackled by real-time, high-speed software technology. For speech recognition, there are still problems in the following areas:

- The phrase recognition rate is approximately 85-90%.
- Word recognition uncertainty will produce many word candidates.
- Increased perplexity¹ is a severe condition for real-time data processing.

Statistical constraints on the input, which take the form of estimates of the a priori probability of occurrence of a particular sequence of words, are used to reduce the perplexity.

In a language processing system, a function which can use syntactic, semantic and pragmatic information to select the most appropriate candidate is necessary. In a spoken language system, it is indispensable to have a technology which uses both background knowledge and information about languages to understand speech content appertaining to a specific field.

The analysis process of the method extracts semantic representations of utterances consisting of illocutionary force and propositional content parts. The illocutionary force part is described in terms of language-independent concepts, and the

propositional content part is described in terms of language-dependent concepts. The method uses feature structures to integrate the description of syntactic, semantic and pragmatic information.

2. Speech recognition

Since conversational speech is normally continuous, with most words running together, the recognition of the elements of continuous speech is necessary. For the recognition of large-vocabulary continuous speech, the problem of attaining high recognition rates must be overcome in order to lessen the burden of language processing. First, as many phonemes as possible must be recognized; the recognition of phrases or sentences then follows.

2.1. Speech recognition by Hidden Markov Modeling

HMM is effective in expressing speech data statistically. Several improvements in HMM phoneme models have been introduced for discrete HMM phoneme recognition (Hanazawa et al., 1989).

The following techniques have been introduced and evaluated:

- (a) duration control;
- (b) separate vector quantization;
- (c) fuzzy vector quantization.

Effective duration control is realized by combining two duration control techniques; i.e. a phoneme duration control for each HMM phoneme model, and a state duration control for each HMM state. The phoneme duration control is carried out by combining HMM output probabilities with phoneme duration histograms obtained from training sample statistics. State duration control is realized by state duration penalties, calculated by modifying the training samples' forward-backward probabilities.

The separate vector quantization techniques for HMM phoneme recognition are useful in reducing VQ distortion. In our case spectral features, spectral dynamic features and energy are quantized separately. At the training stage, the output vector probabilities of these three

¹ Perplexity means the average number of words from which the recognizer has to make a choice at any single point.

codebooks are estimated simultaneously and independently and, at the recognition stage, all the output probabilities are calculated as the product of the output vector probabilities of these codebooks.

2.2. HMM training procedures

HMM training procedures are performed using an extensive training data set (2,620 words) uttered by one male speaker. Phoneme boundaries are specified accurately by the visual examination of spectrogram outputs. The phoneme boundary information is used in training procedures and experiments on the recognition of boundary information. Duration control techniques with three separate codebooks for spectral features, spectral dynamic features, and energy have proved effective in HMM phoneme recognition. For the speaker-dependent case, experiments on the phoneme recognition resulted in a recognition rate of 94.0% for an average of three speakers using the separate codebook and duration control techniques.

The fuzzy VQ technique is effective for parameter smoothing when the number of training samples is insufficient.

3. Integrating speech and language processing

Figure 1 shows a block diagram of a proposed speech translation system which involves speech

recognition, the integrated processing of speech and language, language translation, and speech synthesis.

Since conversational speech is normally continuous with most words running together, recognition of phrases of continuous speech is necessary. Continuous Japanese speech is not as easily segmented into words as English because stem boundaries, bert endings and auxiliary verb endings are not explicit. In Japanese, it is considerably easier to utter a sentence by inserting pauses between phrases. This being the case, the information from the speech recognizer to the language processor is in the form of a phrase lattice.

3.1. Continuous HMM speech recognition using an LR parser

The HMM phoneme models are integrated with the generalized LR predictive parser, as shown in Figure 2. The LR parser was originally developed as a compiler, and then extended to handle arbitrary context-free grammar. An LR parser, guided by an LR table automatically created from context-free grammar rules, proceeds from left to right without backtracking. In the LR parsing mechanism, the next parser action (accept, error, shift, or reduce) is determined by looking up in the LR table the current state of the parser and the next input symbol. This parsing mechanism is valid only for symbolic data and cannot simply be applied to continuous data such as speech.

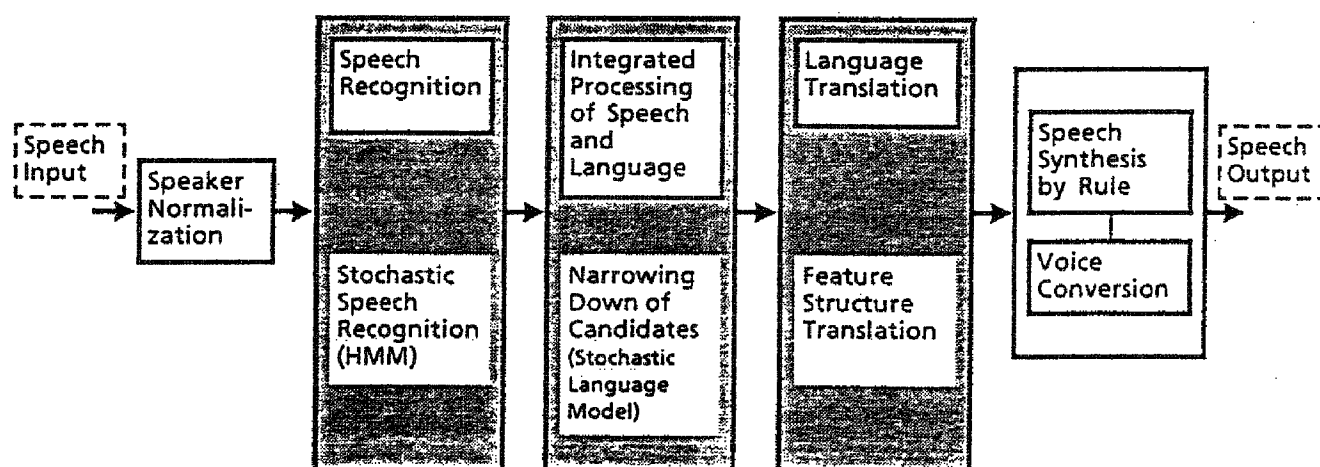


Fig. 1. Block diagram of proposed speech translation interpretation system.

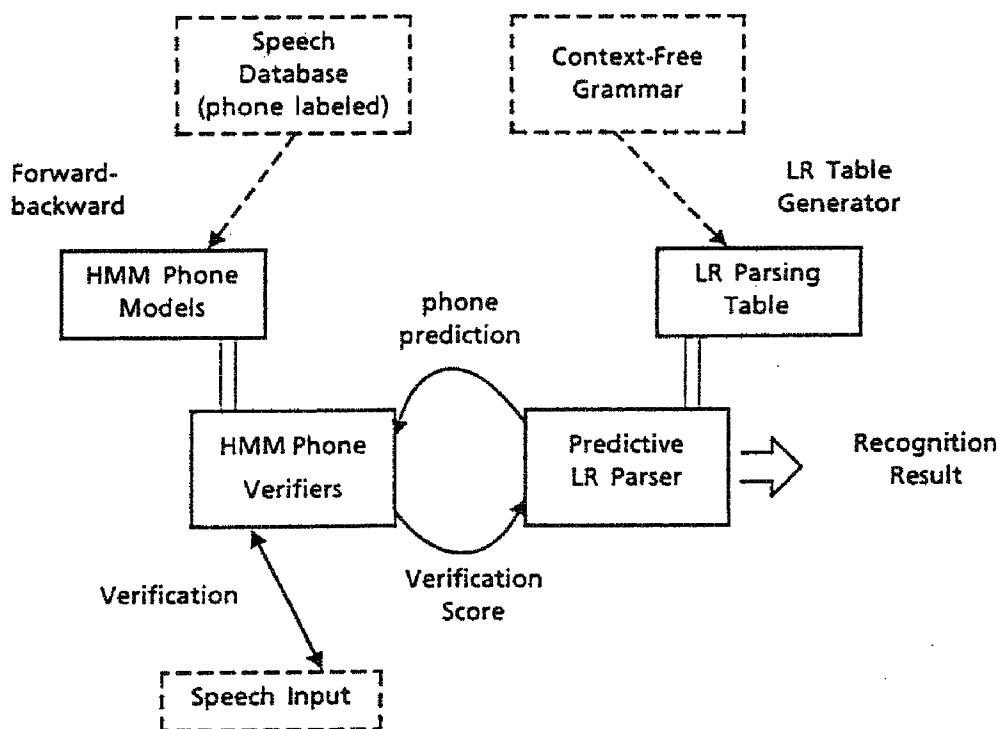


Fig. 2. Schematic diagram of HMM-LR speech recognizer.

In our approach, the LR table is used to predict the next phoneme in the speech input (Kita and Kawabata, 1989). For phoneme prediction, the grammar terminal symbols are phonemes instead of the grammatical category names generally used in natural language processing. That means that a lexicon for the task is embedded in the grammar. The following describes the operation of the system: first, the parser picks up all the phonemes predicted by the initial state of the LR table, and invokes the HMM phoneme models to verify the existence of these expected phonemes. During this time, every possible parsing tree is constructed in parallel. The phoneme verifier (HMM phoneme model) receives a probability array including end point candidates and their probabilities, and updates it using an HMM phoneme probability calculation process (trellis algorithm). This probability array is attached to each node of the partial parsing tree. When the highest probability in the array is below a certain threshold; the parsing tree is pruned and then re-pruned by a beam-searching algorithm with a certain beam width at each phoneme recognition stage. The parsing process stops when the parser

detects an accepted action in the LR table and the end of an utterance.

This integration algorithm is applied to Japanese phrase recognition. The task is the secretarial service of an international conference. Utterances are produced phrase-by-phrase. The syntax of the phrases includes a general syntax structure of Japanese phrases, the perplexity per phoneme of which is about five. Assuming that the average word phoneme length is three, their perplexity is more than one hundred.²

The HMM phoneme models are trained using 5,240 words. The duration control parameters are modified according to the ratio of word utterance speed to phrases. The grammar is designed to cover linguistic expressions common in Japanese. The number of different words is 1,035. For the speaker-dependent case, the phrase recognition rate is 89%.

² In Japanese, a continuous sentence is not definitely separated into words. In calculating perplexity, phoneme perplexity is one of the adequate measure. Word perplexity is assumed by the calculation with the phoneme perplexity and the average word phoneme length.

3.2. Reduction in the number of candidates

In language processing systems, a function is necessary which can use syntactic and semantic knowledge to select the most appropriate candidate. A method which reduces the number of HMM-LR candidates is being studied as part of the analysis of spoken Japanese utterances. In this method, the Japanese *kakari-uke* relationship³ is used (Kakigahara and Morimoto, 1989). This is a kind of semantic relationship between two *bunsetsu*.⁴ About 60 kinds of semantic relationships are defined and attached to each *bunsetsu* in the text database. From this database, a possible *kakari-uke* relationship and its frequency for two *bunsetsu* are extracted. Using this information, only probable *bunsetsu* candidates are selected from the HMM-LR output and the candidates with no *kakari-uke* relationship on other candidates are discarded. Preliminary experiments indicated the usefulness of this method: the number of candidates was reduced to less than one-third of the number of raw candidates.

4. Translation method for spoken dialogues

Hearing the speaker's utterance, the hearer receives communicative signs in addition to a propositional content. According to speech act theory, these signs are classified as illocutionary forces governed by certain felicity conditions (Searle, 1969). Illocutionary forces can be useful to machine translation if propositional content is distinguished from the structure in the utterance analysis. There are many conventional forms for illocutionary forces which concern certain explicit expressions. Japanese sentences in dialogues often have complex sentence final predicate

phrases consisting of main predicates (e.g., verb, adjective, etc.) and combinations of auxiliary verbs and sentence final particles. These phrases are important for expressing intentions.

As the SEMantic feature, the utterance analysis process extracts the values of output phrase structure analysis results, where are semantic representations consisting of surface illocutionary force relationships in terms of language-independent concepts and propositional content parts in terms of language-dependent concepts (Kogure et al., 1989).

Most of the information used in these processes is described in terms of feature structures and is modified by the unification operation on them. Representation formalism using feature structures has various advantages over other representation formalisms. One of the most significant advantages is that they permit integrated descriptions of information from various kinds of information sources such as syntax, semantics and pragmatics.

4.1. Utterance analysis

The utterance analysis process of this method consists of phrase structure analysis and zero pronoun resolution sub-processes. The zero pronoun resolution process takes the output structures and fills in the semantic information corresponding to the zero pronouns by matching pragmatic constraints with dialogue-participant information such as social relationships.

The analysis of spoken utterances in a traditional syntactic-rule-based approach is very likely to require numerous complicated grammatical rules for fragmental expressions and complicated expressions of intention. This makes the maintenance of grammatical rules very hard. A lexico-syntactic approach can resolve the problem. In this approach, a grammar has only a small number of general phrase structure rules corresponding to sets of grammatical principles such as the HEAD and SUBCAT feature principles, and most of the grammatical information must be specified in descriptions of lexical items. It is thus easy to extend a grammar simply by adding new lexical items to the lexicon, or adding new information to existing lexical items.

³ In Japanese, it is well known that one *Bunsetsu* (generally a phrase in English), other than the last *Bunsetsu* of a sentence, has a *Kakariuke* dependency on a unique *Bunsetsu* in a backward position, and that two *Kakariuke* dependency relationships do not cross each other.

⁴ A *Bunsetsu* is a grammatical and phonological unit in Japanese. It consists of an independent-word such as noun, verb or adverb followed by a sequence of zero or more dependent-words such as auxiliary verbs, postpositional particles or sentence final particles.

4.1.1. Head-driven Phrase Structure Grammar for Japanese utterances

In order to implement the unification-based approach, a complement-head grammatical structure based on a version of Head-Driven Phrase Structure Grammar (HPSG) is adopted (Shieber, 1986; Karttunen, 1986). It is represented as a grammatical rule as shown in Figure 3. The principle notions of the grammar are derived from JPSG (Gunji, 1987). The statement consists of two parts, i.e. CFG and equations. CFG is used only to propose a top-down expectation in the analysis. Angled brackets are used to denote a feature structure path, and “=” to denote a token identity relationship between two such structures. The CF production rule is used to limit the application of the feature structure constraint set so as to gain efficiency in the analysis. This set specifies properties of phrase structures consisting of a verbal head and a postpositional complement. For example, the first equation specifies that the HEAD feature of the phrase should be tokenly identical with the HEAD feature of the head. Such constraints permit compositional semantic analysis, gathering pragmatic constraints, etc.

4.1.2. Treatment of sentence final complex predicate phrases

In the registration query task, the illocutionary force relationship set, {“REQUEST (Speaker, Hearer, ACT(H))”, “INFORM (S, H, Proposi-

```
(DEFRULE V-> (P V)    ;; Post-position
(<0 HEAD == <2 HEAD>)
(<1> == <2 SUBCAT FIRST>)
(<0 SUBCAT> == <2 SUBCAT REST>)
(<1 HEAD COH> == <2>)    ;; Category Of Head
(<0 SLASH IN> == <1 SLASH IN>)
(<1 SLASH OUT> == <2 SLASH IN>)
(<2 SLASH OUT> == <0 SLASH OUT>)
(<0 SEM> == <2 SEM>)
(<0 PRAG SPEAKER> == <1 PRAG SPEAKER>)
                        ;; PRAGmatics
(<0 PRAG HEARER> == <1 PRAG HEARER>)
(<0 PRAG SPEAKER> == <2 PRAG SPEAKER>)
(<0 PRAG HEARER> == <2 PRAG HEARER>)
(<0 PRAG RESTR IN> == <1 PRAG RESTR IN>)
                        ;; RESTRiction
(<1 PRAG RESTR OUT> == <2 PRAG RESTR IN>)
(<2 PRAG RESTR OUT> == <0 PRAG RESTR OUT>))
```

Fig. 3. Example of a grammatical rule.

tion)”) is established, and REQUEST calls for an action “INFORMIF (P)” or “INFORMREF (P)” which demands either a yes/no answer to H or a referent identification of H. For example, the representation in EX-1 is of the illocutionary force relationship REQUEST (S, H, INFORMIF (P)). The underlined Japanese words and phrases correspond to the English equivalents. Non-underlined final phrases indicate certain intentions. The utterance in EX-2 is an idiomatic expression.

(EX-1) REQUEST (S, H, INFORMIF (P)):

“tourokuyoushi wa o-mochi
a registration form TOP HON-have
des ka”
COP QUEST⁵
 (“Do (you) have a registration form?”)

(EX-2) INFORM (S, H, P):

“shouchi itashi mashi ta”
understand CONDES POL PAST
 (“(I) understand.”)

In such a predicate phrase, the head constituent stipulates the properties of the complement occurring on the immediate left such as the part of speech, conjugation type, and conjugation form. Such stipulations are easily described in the SUBCAT feature value in the head.

4.1.3. Treatment of zero pronouns

Certain kinds of zero-pronouns can be resolved by using some pragmatic conditions on uses of honorific expressions (Yoshimoto, 1988). As the SEM feature the utterance analysis process extracts the value of an output phrase structure analysis result with some annotations or set of conditions under which the interpretation is valid. For example, as the pragmatic felicity conditions for the use of the honorific prefix “o”-mochi in EX-1, the analysis result has the following annotation:

```
[[relation condescend] [agent ?speaker]
[object ?subject_sem]
[comparative-object ?indirect-object_sem]]
```

⁵ TOP: topic case marker, HON: honorific prefix, COP: copulative verb, QUEST: question particle, CONDES: auxiliary verb for expressing condescending relation, POL: auxiliary verb for expressing politeness, PAST: past auxiliary verb.

The annotation means that the speaker condescends to the person referred to by ?subject_sem, which is the semantic representation of the subject of "mochi".

The descriptions of discourse participants can be given as some components of the dialogue environment. By searching for discourse participants satisfying the annotation, candidates of missing elements in an utterance can be found. In order to obtain such annotations, lexical descriptions have PRAG/RESTRS features which include constraints in terms of RESPECT, CONDESCEND, POLITE, etc.

4.2. Utterance transfer and generation

4.2.1. Utterance transfer

An output of the phrase structure analysis is a semantic representation consisting of an illocutionary force part in terms of language-independent concepts and a propositional content part in terms of language-dependent concepts. So the utterance transfer process treats only the propositional content part.

Propositional contents are represented by recursively defined relationships. Such a relationship consists of a relationship name and its case roles. The transfer process converts such relationships in terms of source language concepts (e.g. concepts in Japanese) into corresponding relationships in terms of target language concepts (e.g. concepts in English). The load of this transfer process is small compared to the load in the traditional syntactic transfer approach because this process does not require many transfer rules

for treating syntactic information.

This non-syntactic transfer process uses a feature structure rewriting system which rewrites input structures according to feature structure rewriting rules. In order to maintain high modularity and flexibility in this transfer process, a rewriting environment has been introduced into the rewriting system and constraints have been attached to each rewriting rule instead of describing rule dependencies explicitly.

The outline of the transfer process is shown in Figure 4. Two different types of representations, propositional contents and the expressions of intention, are translated separately. The analysis result of the sample utterance in Figure 4 is as shown in Figure 5.

4.2.2. Utterance generation

The utterance generation process of this method currently uses illocutionary force expression patterns to generate adequate English sentences. For example, the transfer result of Japanese sample sentence in Figure 4 is:

```
[[relation ga-MODERATE]
 [object [[relation tai-DESIRE]
          [experiencer ?speaker]
          [object [[relation MAKE]
                  [agent ?speaker]
                  ...]]]]]]
```

The generation process outputs "would like to" for the DESIRE relationship provided that it is embedded in the ga-MODERATE relationship. Otherwise, it outputs "want". Such patterns guarantee both a correct translation of illocutionary force and an adequate level of politeness.

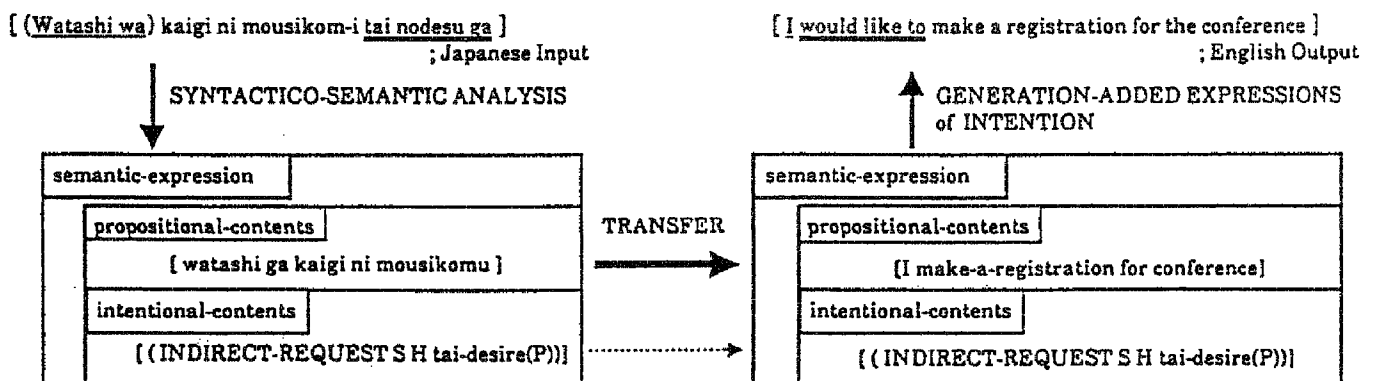


Fig. 4. Outline of dialogue translation method.

```

[[SEM [[RELN ga-MODERATE]
  [OBJE [[RELN tai-DESIRE]
    [EXPR ?X03 [[LABEL *SPEAKER*]]
    [OBJE [[RELN moushikomu-1]
      [AGEN ?X03]
      [SLOC [[PARM ?X02 [ ]]]
      [RESTR [[RELN kaigi-1]
        [OBJE ?X02]]]]]]]]]]]]
[PRAG [[RESTR [[IN [ ]]]
  [OUT [ ]]]]
[SPEAKER ?X03]
[HEARER [[LABEL *HEARER*]]]]]]
('kaigi ni moushikom-i tai nodesu ga' in Figure 4)

```

Fig. 5. Example of analysis result.

5. Experimental speech translation system

5.1. The results of the experimental dialogue translation system NADINE

The experimental translation system has been implemented in Symbolics Common Lisp on a Symbolics 3620, and in Sun Common Lisp on a SUN4. The system consists of three basic modules corresponding to processes in the intention translation method, i.e. analysis, transfer and generation modules.

An analysis module consists of a phrase structure analysis module and a zero pronoun resolution module. The grammar consists of 20 generic phrase structure rules and about 400 lexical entries.

The transfer process consists of the feature structure rewriting system and rules.

The experimental spoken language machine translation system NADINE is applied to 7 Japanese goal-oriented dialogues around registra-

tion for an international conference. These dialogues consist of 138 sentences. The NADINE system obtains adequate English sentences for 126 sentences. Typical translation results are shown in Figure 6.

Very difficult problems remain:

(1) *English NP properties such as definiteness and number.* Almost all Japanese NPs lack information on definiteness and number. In the current NADINE implementation, nominal lexical entries are divided into two classes: entries whose possible referents are unique in the task domain, and other entries.

(2) *English future tense.* Japanese sentences do not have any future tense marker, and there are many sentences to be translated into English future tense sentences in the field.

(3) *Context-dependent idiomatic expressions.* There are expressions which cannot be translated adequately without an understanding of the context of the utterance (Iida et al., 1990), (Yamaoka and Iida, 1990). For example, "yoroshiku onegaishi masu" expresses different deeper level illocutionary forces in different dialogue contexts such as a DECLARATIVE utterance for conforming (an adequate English translation is "thank you") and a DECLARATIVE utterance for a closing section shutting down the dialogue ("good-bye").

5.2. Experiment in spoken language translation

An experimental spoken language translation system (SL-TRANS) has been implemented at ATR. SL-TRANS can recognize Japanese speech input and translate it into English with a synth-

Japanese utterance inputs:	English utterance translation outputs:
A: Moshi moshi	A: Hello.
Sochira wa kaigijimukyoku desu ka	Is that the office for the conference?
B: Hai	B: Yes.
Soudesu	That is right.
A: (Watashi wa) kaigi ni moushikomi tai nodesu ga	A: <u>I would like to make a registration</u> for the conference.
B: (Anata wa) touroku-youshi wa sudeni o-mochi des-you ka	B: <u>Do you already have</u> a registration form?
A: Iie mada desu	A: No. Not yet.
B: Wakarimasita	B: I see.
Soredewa (watashi wa anata ni) touroku-youshi wo o-okuri itasi masu	Then, I will send you a registration form.
;; where parentheses are used to denote missing phrases	;; where double underlining is used to denote supplemented phrases

Fig. 6. Translation examples of the experimental results.

esized English voice output. The system includes a speech recognition system (HMM-LR), a controller of *Bunsetsu* candidates, a machine translation system NADINE, and a speech synthesizer (DECTALK).

Experiments have been conducted with dialogues containing 37 sentences for the conference registration task in a small scale. In these the average number of *Bunsetsu* phrases per sentence was 2.2. Since the HMM-LR phrase recognition was not perfect for the first rank, the top five phrase output candidates were taken and filtered in the *Bunsetsu* filtering process. The average number of sentence candidates was reduced to 2.4. The number of sentences selected correctly through the language analysis was 34.

6. Conclusion

This paper presents a spoken dialogue translation method which mainly handles the expression of intention in each utterance, anaphora resolution, with particular Japanese zero-anaphora, and utterance transfer. An experimental spoken dialogue translation system SL-TRANS combined with HMM-LR parser is also presented.

We are currently designing unification-based translation and generation modules and developing a totally consistent translation system. Our lexico-syntactic parsing method can be applied to many languages besides Japanese. An English-to-

Japanese translation system using an English lexicon is planned.

References

- T. Gunji (1987), *Japanese Phrase Structure Grammar* (Reidel, Dordrecht).
- T. Hanazawa, T. Kawabata and K. Shikano (1989), "Recognition of Japanese voiced stops using Hidden Markov Models", *J. Acoust. Soc. Japan*, Vol. 10, pp. 776-785.
- H. Iida, T. Yamaoka and H. Arita (1990), "Three typed pragmatics for dialogue structure analysis", *Proc. 13th Internat. Conf. on Computational Linguistics*, pp. 370-372.
- K. Kakigahara and T. Morimoto (1989), "A method of *bunsetsu* candidate selection using Kakari-uke semantic relationships", *Proc. Fall Meeting of Acoust. Soc. Japan* (in Japanese), pp. 93-94.
- L. Karttunen (1986), "D-PATR", *Stanford U. Report*, No. CSLI-86-61, 1986.
- K. Kita and T. Kawabata (1989), "HMM continuous speech recognition using predictive LR parsing", *Proc. IEEE ICASSP-89*, pp. 703-706.
- K. Kogure, H. Iida, K. Yoshimoto and T. Aizawa (1989), "A new paradigm of dialogue translation", *Proc. Computer World 89*, Osaka, pp. 135-142.
- J. Searle (1969), *Speech Acts* (Cambridge University Press).
- S.M. Shieber (1986), "An introduction to unification-based approaches to grammar", *CSLI Lecture Notes*, No. 4, pp. 11-36.
- T. Yamaoka and H. Iida (1990), "A method to predict the next utterance using a Four-layered plan recognition model", *Proc. Conf. on European AI*, pp. 726-731.
- K. Yoshimoto (1988), "Identifying zero pronouns in Japanese dialogues", *Proc. 12th Internat. Conf. on Computational Linguistics*, pp. 779-784.

Integration of Speech Recognition and Language Processing in a Japanese to English Spoken Language Translation System

Tsuyoshi MORIMOTO[†], Kiyohiro SHIKANOT^{††}, Kiyoshi KOGURE^{†††},
Hitoshi IIDA[†] and Akira KUREMATSU[†], *Members*

SUMMARY The experimental spoken language translation system (SL-TRANS) has been implemented. It can recognize Japanese speech, translate it to English, and output a synthesized English speech. One of the most important problems in realizing such a system is how to integrate, or connect, speech recognition and language processing. In this paper, a new method realized in the system is described. The method is composed of three processes: grammar-driven predictive speech recognition, Kakariuke-dependency-based candidate filtering, and HPSG-based lattice parsing which is supplemented with a sentence preference mechanism. Input speech is uttered phrase by phrase. The speech recognizer takes an input phrase utterance and outputs several candidates with recognition scores for each phrase. Japanese phrasal grammar is used in recognition. It contributes to the output of grammatically well-formed phrase candidates, as well as to the reduction of phone perplexity. The candidate filter takes a phrase lattice, which is a sequence of multiple candidates for a phrase, and outputs a reduced phrase lattice. It removes semantically inappropriate phrase candidates by applying the Kakariuke dependency relationship between phrases. Finally, the HPSG-based lattice parser takes a phrase lattice and chooses the most plausible sentence by checking syntactic and semantic legitimacy or evaluating sentential preference. Experiment results for the system are also reported and the usefulness of the method is confirmed.

1. Introduction

A spoken language translation system can recognize input speech, translate it to a target language, and output synthesized speech. One of the most important problems in realizing such a system is how to integrate, or connect, speech recognition and language processing. Speech recognition must handle multiple candidates simultaneously to examine all possible hypotheses. This is because its primary objective is to attain a high recognition rate rather than to obtain the exact syntactic structure of each candidate. Language processing, on the other hand, must calculate the semantics as well as the exact syntactic structure of an

input sentence based on some linguistic formalization. Since these tasks are generally time consuming, the number of candidates capable of being treated at the same time is rather low.

When integrating these two processes, an effective and efficient interface should be established.

In the speech recognition process, some linguistic knowledge should be used. Many approaches have been reported, ranging from using rather simple word collocation knowledge (such as bi-gram or tri-gram) to using general context-free grammar^{(1),(2)}. While a context-free grammar approach is more desirable than others because of its generality, a predictive recognition algorithm which can reduce perplexity and attain high recognition accuracy has not been realized.

As for language analysis, many theoretical advances have recently been achieved such as HPSG⁽³⁾ which can formalize various natural language phenomena in a uniform way, and several parsing mechanisms based on these theories were developed. However, it is necessary to develop a parsing mechanism that can handle multiple candidates simultaneously and can select the one which is most plausible.

In this paper, we propose a new method of integrating speech recognition and language processing, which is implemented in the experimental spoken language translation system (SL-TRANS). SL-TRANS⁽⁷⁾ recognizes Japanese speech, translates it to English, and outputs synthesized English speech.

2. An Overview of the Method

The method is composed of three processes (Fig. 1): grammar-driven predictive speech recognition, Kakariuke-dependency-based candidate filtering, and HPSG-based lattice parsing supplemented with a sentence preference mechanism.

Input speech is uttered phrase by phrase (here, phrase means Japanese Bunsetsu*). The speech recog-

Manuscript received January 8, 1991.

Manuscript revised April 9, 1991.

[†] The authors are with ATR Interpreting Telephony Research Laboratories, Kyoto-fu, 619-02 Japan.

^{††} The author is with NTT Human Interface Laboratories, Musashino-shi, 180 Japan.

^{†††} The author is with NTT Basic Research Laboratories, Musashino-shi, 180 Japan.

* A Bunsetsu is a grammatical and phonological unit in Japanese. It consists of an independent-word such as a noun, verb or adverb followed by a sequence of zero or more dependent-words such as auxiliary verbs, post-positional particles or sentence final particles.

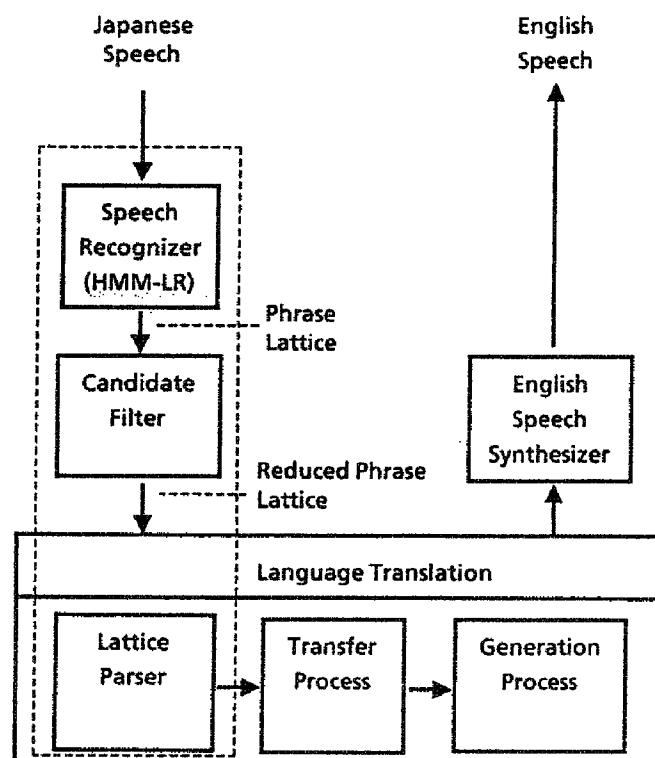


Fig. 1 Configuration of SL-TRANS.

nizer called HMM-LR⁽⁸⁾ takes an input phrase utterance and outputs several candidates with recognition scores for each phrase. Japanese phrasal grammar is used in recognition, and the output candidates are all grammatically well-formed within a phrase. Moreover, it contributes to reduction of phone perplexity, since it is used for phone prediction.

The candidate filter⁽⁹⁾ takes a phrase lattice, which is a sequence of multiple candidates for a phrase, and outputs a reduced phrase lattice. It removes semantically inappropriate phrase candidates by applying the Kakariuke dependency relationship between phrases.

Finally, the HPSG based lattice parser⁽¹⁰⁾, which is the first part of the translation system, takes a phrase lattice and chooses the most plausible sentence by checking syntactic and semantic legitimacy or evaluating sentential preference.

3. Grammar-Driven Predictive Speech Recognition

3.1 HMM-LR

HMM-LR is an integrated speech recognition mechanism, which combines the generalized LR parsing algorithm and the HMM phone model.

A generalized LR parser can handle arbitrary context free grammar⁽⁵⁾. Parsing is guided by an LR table which is created from pre-defined grammar, and proceeds left-to-right without backtracking. If syntactical ambiguities are encountered, it splits the parsing

stacks and proceeds to construct multiple partial syntax trees in parallel.

In HMM-LR, the terminal symbols in the grammar are phones, that is, lexical items are embedded in the grammar. The parser runs in a predictive way. It sees which phones should come next by consulting the LR table, then verifies their existence by comparison with the corresponding HMM phone models. If several phones are predicted, it verifies their existence and keeps all possible parsing trees in split stacks. In this process, the probabilities for each partial parsing tree are calculated and only those with a higher probability are kept, using the beam search technique. The recognition process proceeds until the end of the input, and those which are accepted successfully and have a higher probability score are output as the final candidates. Some recognition results are shown below. Numbers in parenthesis are normalized probability scores described in Sect. 3.2.

[EX. 3-1]

Input (sochira-wa) [is this]

- | | | |
|----|---------------|------------|
| 1. | sochira-wa | (0.949367) |
| 2. | itsu-kara-wa | (0.012658) |
| 3. | sochira-kara | (0.012658) |
| 4. | sochira-to-wa | (0.012658) |
| 5. | sochira-o | (0.012658) |

Input (kaigizimukyoku-desu-ka) [the Conference office ?]

- | | | |
|----|------------------------|------------|
| 1. | kaigizimukyoku-desu-ka | (0.935724) |
| 2. | hachizyuu-roku-desu-ka | (0.064276) |

3.2 Normalization of Recognition Score

HMM-LR calculates a phrase candidate recognition score by multiplying probabilities for each constituent phone and dividing it by the phrase length. Scores determined in this way generally fluctuate somewhat. This fluctuation makes it hard to compare scores of candidates for different phrases in the succeeding processes. Such comparison does not occur while only the top candidates of the each phrase are treated, but does occur when the second or the lower ranked candidates must be tested to make alternative sentential hypotheses according to their recognition scores.

To enable such comparison, the original HMM-LR score is normalized so that it indicates the absolute probability of correctness of each candidate.

From the preliminary recognition result, it was seen that the probability of correctness p of a phrase candidate has strong correlation with rank r and the distribution v of original HMM-LR score (Fig. 2) :

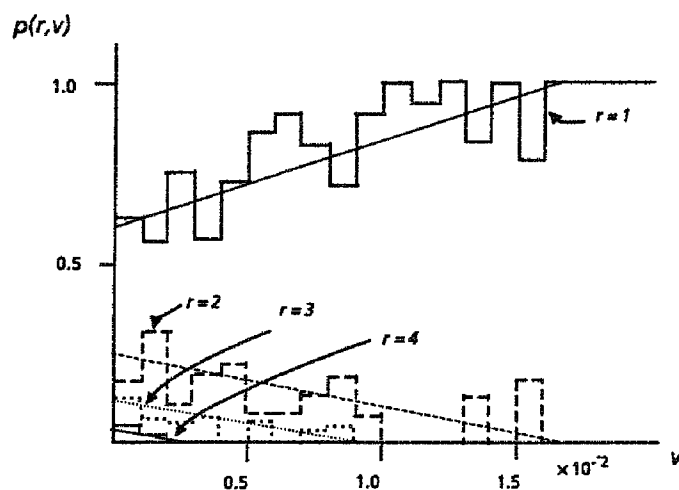


Fig. 2 Probability of correctness.

when the rank is higher or the distribution is larger, the probability is higher. Then, this probability distribution function is written as $p(r, v)$ (here, $\sum_r p(r, v) = 1$).

Once this distribution function is obtained, the probability of correctness of a certain candidate can be calculated with it. In the actual system described in Sect. 6, $p(r, v)$ for each r is approximated by straight lines indicated in Fig. 2.

4. Candidate Filtering Based on Inter-Phrase Kakariuke Dependency

If all phrase candidates output from HMM-LR are transferred directly to the parser, its processing time will explode, because the number of sentential hypotheses becomes quite large and exact linguistic analysis generally takes much time as described before.

The purpose of the candidate filtering process is to eliminate implausible candidates and to lighten the load of the parser. This elimination is done by checking the inter-phrase Kakariuke dependency relationship between phrase candidates.

4.1 Kakariuke Dependency Relationship

Kakariuke dependency is a kind of semantic relationship between two phrases. Typical examples are shown in column [A] of Table 1. We call a former phrase a source and a latter phrase a destination.

In Japanese, one phrase, except the last phrase of a sentence, has a Kakariuke dependency relationship with a unique phrase on the right-hand-side, and no two Kakariuke dependency relationships in one sentence cross each other. An example of a Kakariuke dependency structure in a sentence is shown in column [B] of Table 1.

Table 1 Example of Kakariuke dependency relationship.

	Example	
	Case-filler to Predicate	sankahi-o registration-fee okuru send
[A] Kakariuke relationship between two phrases	Noun to Noun	kaigi-no of a conference sankahi registration-fee
	Adverb to Predicate	sikyuu immediately okuru send
[B] Kakariuke structure in a sentence	kaigi-no of a conference sankahi-o registration-fee sikyuu immediately okuru send	

4.2 Candidate Filtering

First, a sequence of phrase candidates which satisfies the Kakariuke dependency relationship is selected from the original phrase lattice. There could be several such sequences, then this selection procedure is repeated until finding the n -best (or all) of them. Finally, all sequences are compiled again in lattice form to avoid unnecessary duplications of the same phrase candidate in different sequences. Another reason is that the structure is more suitable for the next process (the lattice parser).

[1] Selection of sequences of candidates satisfying Kakariuke dependency

Several algorithms which select phrase sequences from a phrase lattice based on Kakariuke dependency relationship have been proposed^{(11),(12)}. They are based on the breadth-first search algorithm, and are effective when many Kakariuke relationship ambiguities exist. We adapted a more simplified method based on depth-first search. It is still effective and efficient if the domain is limited as described below.

Selection is executed as follows.

(Step 1) Choose one phrase candidate which has the highest recognition score among all the candidates in the phrase lattice.

(Step 2) Find a destination candidate from the right-hand-side phrases. If several destinations are found, select the one which has the highest matching score (described later).

(Step 3) Taking the candidate found in Step 2 as a source, repeat Step 2 until the end of the sentence.

(Step 4) If there are phrases from which no candidate is chosen, take the right-most phrase from them. Repeat Step 5 until no such phrase remains.

(Step 5) Find one candidate which will be a source of one of the already selected candidates on the right-hand-side. The Kakariuke relationship it will newly make must not cross the Kakariuke relationships

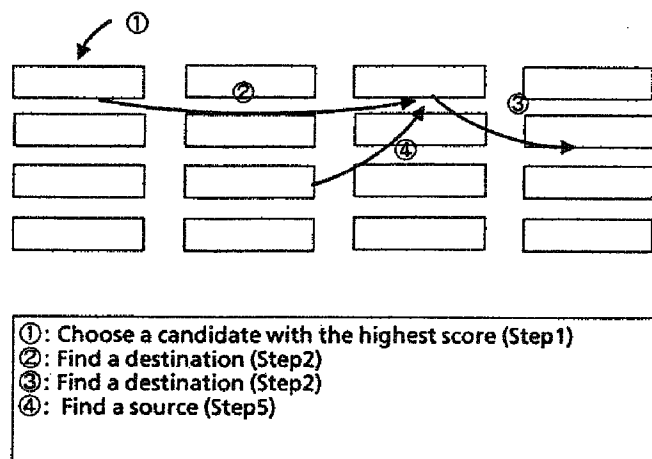


Fig. 3 Kakariuke selection.

Source	Destination	Frequency
kaigi-ni	moushikommu	10
kaigi-ni	sankasuru	5
:	:	:
:	:	:

Fig. 4 Kakariuke dictionary.

already made. If several candidates are found, select the one which has the highest matching score.

An example is illustrated in Fig. 3.

[2] Kakariuke dictionary

What sort of phrases could be in Kakariuke relationship is defined in the Kakariuke dictionary. Figure 4 shows an example of the contents of the dictionary. In each entry, a source and a destination phrase in a Kakariuke relationship are defined. The dictionary is compiled from the analysis of the ATR corpus⁽¹³⁾. The corpus contains a large number of various conversational texts gathered by simulation. In the dictionary, the appearance frequency of each combination in the corpus is also defined. In a sense, the content of the dictionary is domain-dependent.

[3] Checking the Kakariuke relationship between two candidates and evaluating a matching score

Checking whether two candidates are in a Kakariuke relationship is performed as follows:

(1) In a Kakariuke relationship between two phrases, the dependent words in the former phrase indicate the semantic relationship to the latter phrase. Then, the matching is executed between "an independent word and dependent words" of the former phrase and "an independent word" of the latter phrase.

(2) The same combination of phrases as the output candidates from HMM-LR are not always found in the dictionary. Then the expression levels for each independent word, as described below, are defined:

- (a) surface expression: a word exactly as it appeared.
- (b) standard expression: a standard word hav-

ing the same meaning in the target domain.

(c) semantic feature expression: semantic feature for a word.

All combinations of two candidates are examined using (a), (b) or (c) in this order. For this sake, three (surface-, standard-, semantic feature-) level Kakariuke dictionaries are prepared in the system.

(3) If several candidates are found in the same level, the one having the highest matching score, which is calculated according to the following equation, is selected.

$$K(X, Y) = F(X, Y) - w_1 \times D(X, Y) + w_2 \times S(Y) \quad (1)$$

where

$K(X, Y)$: matching score between phrase X and Y

$F(X, Y)$: appearance frequency defined in the dictionary

$D(X, Y)$: distance between X and Y in the input data

$S(Y)$: speech recognition score

w_1, w_2 : weights determined experimentally

4.3 Detection of Speech Recognition Failure

If all candidates for a phrase do not have a Kakariuke relationship with other phrases, it could be assumed that the speech recognition of the phrase has failed. When such failure is detected, the system informs the user that the phrase might have been recognized erroneously, and asks him to enter it again (the current system only allows re-entering through the keyboard, since it handles only a certain specific user's speech which has been recorded beforehand).

5. HPSG-Based Lattice Parser

5.1 Parsing Mechanism

The parser is essentially based on HPSG (and its Japanese version JPSG⁽⁴⁾) grammar formalism and lexico-syntactic framework. In each lexical item, abundant syntactic, semantic and pragmatic information is defined. For each context free grammar rule, constraints that the elements (left-hand-side and right-hand-side of the rule) should satisfy are defined. When applying the rule, input constituents are examined to see whether they satisfy these constraints. Using these schema, grammatically ill-formed, semantically illegitimate, or pragmatic inappropriate sentential candidates can be checked out.

The parsing mechanism itself is based on the active chart parsing algorithm⁽⁶⁾ because it has several advantages in dealing with a lattice input. A lattice structure can be mapped to a chart in a natural way; a

phrase candidate is mapped to an edge in a chart. The agenda control mechanism of the chart parser enables preferential parsing of plausible hypotheses, which is essential to avoid exhaustive search of large search space such as a lattice input. Throughout parsing, the edge with higher plausibility is more readily chosen. The plausibility is determined primarily by taking the speech recognition score into consideration.

5.2 Sentence Preference

Parsing finishes when a certain number of parsed results have been created. If there is more than one result, the one with the highest preference score, which is calculated according to the following equation, is selected.

$$P(X) = a1 \times S(X) - a2 \times Nt(X) - a3 \times Nu(X) \quad (2)$$

where

$P(X)$: preference score of sentence X

$S(X)$: sentential recognition score obtained as the product of all constituent phrase speech recognition scores

$Nt(X)$: number of syntax tree nodes

$Nu(X)$: number of unfilled obligatory elements

$a1, a2, a3$: weights determined experimentally

Nt and Nu reflect the heuristics that a simpler or more complete sentence is more plausible. The evaluation is executed after resolving ellipses in the sentence, since words which are easily inferable from the context, such as "watashi (I)" or "anata (you)", are seldom uttered in spoken Japanese. (the method of resolving such ellipses is reported in Ref. (14))

An example of sentence preference is indicated below.

[Ex. 5-1]

Input :

(1-1) soredewa (2-1) saremasu
then (someone) do (something) / (something) is done[†]

(1-2) sureba (2-2) shitsureishimasu
if (someone) do (something) goodbye

Evaluation of Preference :

Of these, the combination of (1-1) and (2-2) has the fewest nodes and unfilled obligatory elements and thus, is selected.

Output :

soredewa shitsureishimasu
then goodbye

6. Experiment Result

6.1 Experiment Condition

Experiments have been conducted for two dialogues of "inquiry about an international conference registration". The number of utterances of each dialogue is 18 and 19. The average numbers of phrases per utterance are nearly the same, and is 2.2 on average (the minimum is 1 and the maximum is 6).

Input sentences are those which are spoken by a male announcer and uttered phrase by phrase. The speech data is sampled at 12 kHz, pre-emphasized by $(1-0.97z^{-1})$ and windowed using a 256-point Hamming window every 3 msec. Then a 12-order LPC analysis is carried out. A code book of 256 LPC spectrum envelope is generated from 216 phonetically balanced words. The PWLR is used as LPC distance measure for vector quantization. Input for HMM-LR is a sequence of vector quantized codes.

Other experiment conditions are shown in Table

Table 2 Experiment conditions.

HMM-LR	<ul style="list-style-type: none"> • Number of Words: 532 words • Parameter: <ul style="list-style-type: none"> • Beam Width: 16
Candidate Filter	<ul style="list-style-type: none"> • Number of Dictionary Entries: <ul style="list-style-type: none"> • Surface-Level: 6536 • Standard-Level: 4558 • Semantic Feature-Level: 2431 • Weights in Eq. (1): <ul style="list-style-type: none"> • $w1$: 1.0 • $w2$: 5.0
Lattice Parser	<ul style="list-style-type: none"> • Number of Words: 437 words • Weights in Eq. (2): <ul style="list-style-type: none"> • $a1$: 1.0 • $a2$: 0.1 • $a3$: 1.0

Table 3 Experiment results.

HMM-LR	<ul style="list-style-type: none"> • Phrase Recognition Rate: <ul style="list-style-type: none"> • The 1st Rank: 83% • Within the Top 5 Ranks: 93% • Utterances Recognized Correctly Within the Top 5 Candidates: 31 • Average Number of Output Candidates / Phrase: 4.6
Kakariuke Filter	<ul style="list-style-type: none"> • Average Number of Selected Candidates / Phrase Without Asking: 2.5 • Number of Asked Phrases <ul style="list-style-type: none"> • Correctly Recognized: 16 • Incorrectly Recognized: 1
Lattice Parser	<ul style="list-style-type: none"> • Number of Input Lattices Including Correct Sentential Candidates: 32 • Average Number of Sentential Candidates / Lattice: 5.8 • Number of Lattices Parsed Correctly: 32

[†] A italic line indicates an explanation of a previous line.

2.

6.2 Result

Experiment results are summarized in Table 3.

HMM-LR usually outputs the top five candidates. However, the average number output actually is 4.6. The phrase recognition rate is 93% for the top five ranks, in which 31 correct sentential candidates are included.

The candidate filter suspects the correctness of 17 phrases and asks the user to re-enter them (however, 16 phrases are correct). The average number of candidates for phrases not asked is reduced to 2.5. Most of the misjudgements are due to the lack of corresponding entries in the Kakariuke dictionary.

Finally, 32 sentences out of 37 are selected and parsed correctly by the lattice parser. An input utterance to the parser is composed of 5.8 sentential candidates on average.

The machines used for execution of the HMM-LR, the candidate filter and the lattice parser are Alliant, VAX station and SUN-4, and their average processing times are about 25 sec., 8 sec. and 80 sec., respectively. From this result, we can see the processing efficiency of the candidate filter is very high.

6.3 Discussion

The following characteristics are seen in the results of the experiment.

(1) Though HMM-LR attains high recognition accuracy, most of the sentential candidates which would be generated by simply combining all the candidates are grammatically or semantically ill-formed, because HMM-LR only uses phrasal grammar. Moreover, short words such as post-positional particles, most of which work as functional words in a sentence, tend to be recognized erroneously as indicated in [Ex. 6-1].

[Ex. 6-1]

Input Utterance :

namae-wa

name-TOP[†]

Recognition Results :

namae-ga

name-NOM

nama-e-wa

seven-to-TOP

namae-ya

name-too

There are few cases in which HMM-LR does not output any correct candidate at all as [Ex. 6-2]. This is caused by the left-to-right and beam search parsing algorithm in the HMM-LR. If erroneous recognition occurs at an early stage, it might result in a catas-

trophic error.

[Ex. 6-2]

Input Utterance :

wakarimashita

(I) see

Recognition Results :

arimashita

(there) was

narimashita

(it) became

(2) Some of the ungrammatical sentential candidates and many of the semantically inappropriate candidates are detected and eliminated by the candidate filter as [Ex. 6-3].

[Ex. 6-3]

Input Utterance :

tourokuyoushi-wa

registration-form-TOP

omochi-deshou-ka

have-POL-Q

Output from HMM-LR :

(1-1) tourokuyoushi-wa

registration-form-TOP

(2-1) omochi-deshou-ka

have-POL-Q

(1-2) tourokuyoushi-ga

registration-form-NOM

(2-2) kugatsu-deshou-ka

september-is/POL-

Q

(1-3) touroku-shi-ta

registered

(2-3) omoi-mashou-ka

think-POL-Q

Kakariuke dependency check at the Filter :

(1-2) cannot be a source of any candidates (predicates) for the second phrase because its case marker is nominative.

(1-3) can logically be a source modifying (2-2), but such Kakariuke does not appear in the Kakariuke dictionary.

(2-3) cannot be a destination of any candidates for the first phrase.

Output from the Filter :

tourokuyoushi-wa

registration-form-TOP

omochi-deshou-ka

have-POL-Q

However, some semantically inappropriate candidates are not detected by the filter, because its algorithm is simplified and does not see n -wise ($n > 2$) relationships between phrases such as a double occurrence of a certain case phrase for one predicate. For example, there are two accusative phrases (marked by "o") in [Ex. 6-4].

[Ex. 6-4]

Output from the Filter :

gozyuusho-o

address-ACC

onamae-o

name-ACC

onegaishimasu

please

[†] Symbols are used for grammatical markers as follows.

NOM: nominative, ACC: accusative, DAT: dative, TOP: topical, POL: polite, INT: intentional, MOD: moderate, COMP: complementizer, Q: question, DES: desire

(3) Many of the sentential syntactic errors are left to the parser. Incomplete sentences which do not end with the proper final form are sometimes generated by HMM-LR and are not detectable by the Kakariuke-dependency filter. Such errors are all detected by the parser.

[Ex. 6-5]

Input Utterance :

namae-wa shimizutarou-desu

name-TOP Tarou Shimizu-is

Recognition Error Detected by the Parser :

namae-wa shimizutarou-de

name-TOP Tarou Shimizu-is, and

There are several sentences composed of only one phrase in the experiment dialogues. Some candidates for them with the same kind of syntactical error are also detected by the parser.

(4) Recognition errors which result in alteration, deletion or insertion of some post-positional particles or sentence-final particles are hard to detect ([Ex. 6-6]). Sentences including such errors are syntactically and even semantically correct. Some contextual processing should be introduced to detect such errors.

[Ex. 6-6]

Input Utterance :

kaigi-ni moushikomi-tai-no-desu-ga

conference-DAT apply for-DES-COMP-
POL-MOD

Recognition Result :

kaigi-ni moushikomi-tai-no-desu-ka

conference-DAT apply for-DES-COMP-POL-Q

7. Conclusion

In this paper, we proposed a method to connect speech recognition and language processing in stepwise fashion. In the speech recognition stage, the HMM phone model and LR parsing mechanism are combined and phone level prediction is realized. In the next step, the candidate filtering based on the Japanese inter-phrase Kakariuke dependency eliminates the inappropriate candidates. Finally, the lattice parser selects the most plausible sentence by checking syntactico-semantic legitimacy or evaluating sentence preference. The experiment results show the usefulness of the method.

Points to be studied in future would be as follows.

(1) To evaluate for more extensive input data.

(2) To evaluate for a non-specific user's speech. In particular, evaluation of the probability distribution function described in Sect. 3.2 might be needed, because it is derived from the recognition results of a specific user's speech.

(3) To improve performance of each component, in particular that of the parser.

At the same time, we will make efforts to enhance the system's total acceptance rate by not only improving the abilities of each component but also by integrating them to work more cooperatively.

Acknowledgement

The authors would like to thank all the members of the ATR Interpreting Telephony Research Laboratories for their collaboration and contributions to the various parts of the research.

References

- (1) Ney H.: "Dynamic Programming Speech Recognition Using a Context-Free Grammar", Proc. ICASSP 87, pp. 69-72 (1987).
- (2) Nakagawa S.: "Spoken Sentence Recognition by Time-Synchronous Parsing Algorithm of Context-Free Grammar", Proc. ICASSP 87, pp. 829-832 (1987).
- (3) Pollard C. and Sag I.: "Head Driven Phrase Structure Grammar"; An Informal Synopsis, CSLI Report No. CSLI-87-79 (1987).
- (4) Gunji T.: "Japanese Phrase Structure Grammar", D. Reidel (1987).
- (5) Tomita M.: "Efficient Parsing for Natural Language: A Fast Algorithm for Practical Systems", Kluwer Academic Publishers (1986).
- (6) Kay M.: "Algorithm Schemata and Data Structures in Syntactic Processing", Technical Report CSL-80-12, Xerox PARC (1980).
- (7) Morimoto T., Iida H., Kurematsu A., Shikano K. and Aizawa T.: "Spoken Language Translation: Toward Realizing an Automatic Telephone Interpretation", Proc. Info-Japan 90, pp. 553-560 (1990).
- (8) Kita K., Kawabata T. and Saito H.: "HMM Continuous Speech Recognition Using Predictive LR Parsing", Proc. ICASSP 89, pp. 703-706 (1989).
- (9) Kakigahara K. and Morimoto T.: "A Method of Bunsetsu Candidate Selection Using Kakariuke Semantic Relationship", Proc. ASJ Fall Meeting (1989).
- (10) Nagata M. and Kogure K.: "HPSG-Based Lattice Parser for Spoken Japanese in a Spoken Language Translation System", Proc. ECAI 90, pp. 461-466 (1990).
- (11) Ozeki K.: "A Multi-Stage Decision Algorithm to Select Optimum Bunsetsu Sequences Based on Degree of Kakariuke-Dependency", Trans. IEICE, J70-D, 3, pp. 601-609 (March 1987).
- (12) Matsunaga S., Sagayama S., Homma S. and Furui S.: "A Continuous Speech Recognition System Based on a Two-Level Grammar Approach", Proc. ICASSP 90, pp. 589-592 (1990).
- (13) Ehara T., Ogura K., Morimoto T.: "ATR Dialogue Database", Proc. ICSLP 90, pp. 1093-1096 (1990).
- (14) Dohsaka K.: "Identifying the Referents of Zero-Pronouns in Japanese Based on Pragmatic Constraint Interpretation", Proc. ECAI 90, pp. 240-245 (1990).



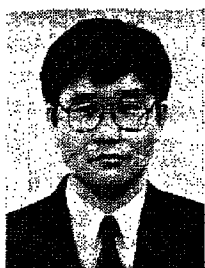
Tsuyoshi Morimoto received the B. E. and M. E. degrees in Electronics Engineering from Kyushu University, Fukuoka, Japan, in 1968 and 1970, respectively. In 1970, he joined the Electrical Communication Laboratories of NTT. He was engaged in the research and development of operating system and database retrieval. Since 1987, he has been with ATR Interpreting Telephony Research Laboratories, Kyoto, Japan. He

is currently the Head of the Knowledge and Data Base Department. His research interests are integration of speech recognition and language processing, and natural language understanding. He is a member of the Information Processing Society of Japan.



Kiyohiro Shikano received the B. S., M. S., and Ph. D. degrees in electrical engineering from Nagoya University in 1970, 1972 and 1980, respectively. From 1972, he has been working at NTT Laboratories, where he has been engaged in speech recognition research. He is currently the Head of Speech Information Processing Department at NTT Human Interface Laboratories, where he is managing the research of speech recognition and speech coding. During 1984-86, he was a visiting scientist in Carnegie Mellon University, where he was working on distance measures, speaker adaptation by codebook mapping, and statistical language modeling. During 1986-1990, he was the Head of Speech Processing Department at ATR Interpreting Telephony Research Laboratories, where he was directing speech recognition and speech synthesis research for interpreting telephone systems. He received the Yonezawa Prize from IEICE in 1975. He is a coauthor of "Speech, Auditory and Neural Network Models" (Series of Neuro Science & Technology, Ohm Press). He is a member of the Institute of Electrical and Electronics Engineers, Information Processing Society of Japan, and the Acoustical Society of Japan.

tion and speech coding. During 1984-86, he was a visiting scientist in Carnegie Mellon University, where he was working on distance measures, speaker adaptation by codebook mapping, and statistical language modeling. During 1986-1990, he was the Head of Speech Processing Department at ATR Interpreting Telephony Research Laboratories, where he was directing speech recognition and speech synthesis research for interpreting telephone systems. He received the Yonezawa Prize from IEICE in 1975. He is a coauthor of "Speech, Auditory and Neural Network Models" (Series of Neuro Science & Technology, Ohm Press). He is a member of the Institute of Electrical and Electronics Engineers, Information Processing Society of Japan, and the Acoustical Society of Japan.



Kiyoshi Kogure received the B. S. and M. S. degrees in electrical engineering from Keio University in 1979 and 1981, respectively. From 1981, he has been working at NTT Laboratories, where he has been engaged in natural language processing research. He is currently a senior research scientist of Information Science Research Laboratory at NTT Basic Research Laboratories, where he is researching natural language understanding. During 1986-1990, he was a member of Natural Language Understanding Department at ATR Interpreting Telephony Research Laboratories, where he was researching unification-based natural language processing for interpreting telephony systems. He is a member of the Information Processing Society of Japan, the Acoustical Society of Japan, and the Association for Computational Linguistics.

ing. During 1986-1990, he was a member of Natural Language Understanding Department at ATR Interpreting Telephony Research Laboratories, where he was researching unification-based natural language processing for interpreting telephony systems. He is a member of the Information Processing Society of Japan, the Acoustical Society of Japan, and the Association for Computational Linguistics.



Hitoshi Iida received the B. S. and M. S. degrees in Mathematics from Waseda University, Tokyo, Japan, in 1972 and 1974, respectively. In 1974, he joined the Electrical Communication Laboratories of NTT. He was engaged in research of natural language processing and machine translation. Since 1986, he has been with ATR Interpreting Telephony Research Laboratories, Kyoto, Japan. He is currently the Head of the

Natural Language Processing Department. His research interests are dialogue understanding and its application to speech recognition. He is a member of the Information Processing Society of Japan, Japanese Society for Artificial Intelligence, Association for Computational Linguistics, AAI, and Japanese Society for Cognitive Science.



Akira Kurematsu received the B. E. degree in Electrical Communication Engineering from Waseda University, Tokyo, Japan in 1961. In 1961, he joined the Research and Development Laboratories of KDD. He was engaged in research of pattern recognition, speech recognition, speech synthesis, speech coding and intelligent terminal systems. In 1971, he received Ph. D. degree from Waseda University. In 1983, he was

appointed the Deputy Director of KDD R&D Labs. Since 1986, he has been with ATR Interpreting Telephony Research Laboratories, Kyoto, Japan. He is currently the President of the Labs. He was a part-time invited lecturer at Chuo University from 1983 to 1987, and at Osaka University in 1990. Dr. Kurematsu has published many invited papers, book chapters and technical reports. He is a member of the Information Processing Society of Japan, the Acoustical Society of Japan, Japanese Society for Artificial Intelligence and IEEE.

RECENT ADVANCES IN SPEECH RECOGNITION

Sadaaki Furui

NTT Human Interface Laboratories
Musashino-shi, Tokyo, 180 Japan

ABSTRACT

This paper introduces recent research activities on speech recognition, ranging from acoustic processing to linguistic processing, at NTT (Nippon Telegraph and Telephone Corporation) Laboratories. These include the proposal of hierarchical Δ cepstral parameters and Δ LSP parameters, a new method of utilizing pitch information, automatic speaker adaptation techniques, robust HMM phoneme models, new training algorithms for neural networks, linguistic processing using syntactic and semantic knowledge, implementation of prototype continuous speech recognition systems, and an efficient text-independent speaker recognition algorithm.

1. INTRODUCTION

Speech recognition technology has recently made remarkable progress in various aspects, especially in speaker-independent continuous speech recognition [Mariani, 1989, Furui, 1989a]. This paper discusses recent advances in speech recognition technology, which are closely related to advanced AI, signal processing, statistical modeling and various other technologies. This paper, however, is not intended to be a comprehensive review of speech recognition covering all of the recent technologies. Rather, it is intended to outline major technologies being investigated at NTT Laboratories.

The most important target of the present speech recognition research is to achieve speaker-independent, large-vocabulary, continuous speech recognition systems. In order to bring forth these systems, various kinds of problems must be solved. Figure 1 shows the principal structure of the continuous speech recognition systems, as well as major problems under investigation.

Flow of the recognition process is as follows. First, endpoints of speech periods are detected, typically using the short-time energy level of input speech. Then, the speech wave is converted into a time sequence of feature parameters through spectral analysis.

Cepstral and Δ cepstral coefficients, representing instantaneous and dynamic features of spectra, are widely used. We recently evaluated two new methods: a method of using hierarchical dynamic features (Subsection 2.1) and a method of using LSP (line spectrum pair) and Δ LSP parameters (Subsection 2.2). We have also proposed a new method of using prosodic features (Subsection 2.3), which have hardly been successfully used in speech recognition. The prosodic features are useful for recognizing several words that are phonetically similar and that have different accent patterns.

Although a high speaker-independent recognition accuracy can be obtained by training the recognition system using utterances from a large number of speakers, this method has a performance limitation. It is therefore necessary to introduce techniques of automatically adapting the recognition system to a new speaker (Subsections 3.1, 3.2 and 3.3).

Although word templates or word models are convenient to use in isolated word recognition, units smaller than words are essential for recognizing large-vocabulary, continuous speech. It is also important for so-called co-articulation effects to be considered when creating these units. We have proposed an efficient method of creating context-dependent phoneme units using a clustering technique (Subsection 4.1). We have compared several representations of HMM phoneme models, from the viewpoint of robustness against variations in speaking styles, such as the differences between word-by-word utterances, phrase-by-phrase utterances, and continuous speech (Subsection 4.2).

Although neural-network phoneme modeling has various new possibilities, it has not yet surpassed the performances of the HMM (hidden Markov model)-based methods. New methods of training neural networks have been investigated to stabilize performance (Subsection 4.3). Phoneme recognition results obtained using these techniques and a word dictionary are then combined to recognize words.

The last and the most important stage of the continuous speech recognition systems is the linguistic processing stage, where

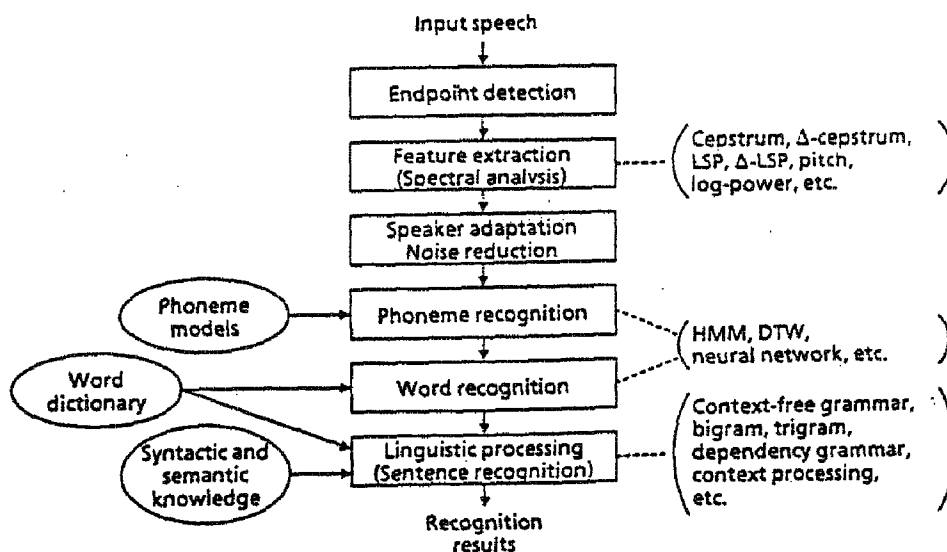


Fig.1. Principal structure of continuous speech recognition systems.

syntactic and semantic knowledge are used. Local language modeling, such as transitional network grammar, bigram- or trigram-based grammar modeling, has recently been used in many recognition systems. How to combine these methods with a global syntax such as case grammar is one of the important research subjects (Subsection 5.1).

Applications of continuous speech recognition can be classified into two categories; voice-input word processors (dictation systems), and various information services, such as guidance, reservation, and order receiving services using natural conversational speech. A dictation prototype system and an experimental dictation system, which we recently implemented, will be introduced in Subsections 5.2 and 5.3. We have also implemented a word spotting hardware into a dialog system (Subsection 5.4).

Speaker recognition using speaker-specific information in speech waves is also an important research subject; it is expected to be used in various applications such as security control in the future (Section 6). Section 7 will be devoted to describing future research topics.

II. FEATURE EXTRACTION

2.1 Hierarchical Spectral Dynamics

The effectiveness of a dynamic spectral feature set, namely, that includes Δ cepstrum and Δ power, to improve recognition accuracy, especially in speaker-independent recognition, has been confirmed [Furui, 1986], and these parameters are now widely used. The parameters are defined as first-order regression coefficients for short-time sequences of cepstral coefficients and logarithmic energy, respectively. They are usually used in combination with conventional cepstral coefficients and logarithmic energy. The length of the parameter sequence for regression analysis is usually set empirically between 40 ms and 100 ms.

Recently we proposed a recognition method that uses hierarchical spectral dynamic features extracted over multiple time lengths (Fig. 2), and showed the effectiveness of these features in phoneme recognition and isolated word recognition [Furui, 1990a]. Δc_l and Δp_l in Fig. 2 indicate regression coefficients calculated over l frames (7 frames = 56 ms, 21 frames = 168 ms) for cepstrum and log-energy, respectively. The speaker-independent isolated word recognition experiments were performed using a vocabulary of 100 Japanese words.

In this method, input speech is quantized by word-specific codebooks created as subsets of a universal codebook (Fig. 3). When VQ distortion is used for word identification, a low recognition error rate of 3.6% is achieved. When VQ distortion is used for preprocessing, that is, pre-selection of candidate words, the number of candidates for each input utterance is reduced to 1% of the vocabulary, with no increase in the error rate. Phoneme recognition experiments were also performed for the consonants /b/, /d/ and /g/ in a large vocabulary of isolated words uttered by one male speaker. Using the proposed recognition method, the high recognition accuracy of 98-99% was obtained. Since the VQ-distortion method does not require time alignment, it has the advantages of less computation and ease of parallel processing.

2.2 LSP Dynamic Features

LSP frequency representation of a speech signal was introduced as an alternative linear predictive coding (LPC) representation for the purpose of maintaining voice quality at smaller bit rates [Itakura, 1975]. This representation functions in the frequency domain, and various researchers have made use of this representation in various speech applications, such as speech coding, synthesis, and recognition. It has been reported that LSP-based distance measures lead to better recognition performance than conventional distance measures [Paliwal, 1990].

We have introduced distance measures based on the linear

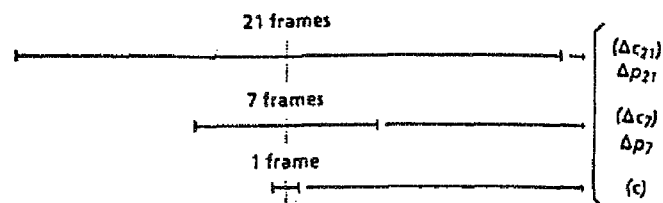


Fig. 2 - Structure of feature vectors for the hierarchical dynamic feature method.

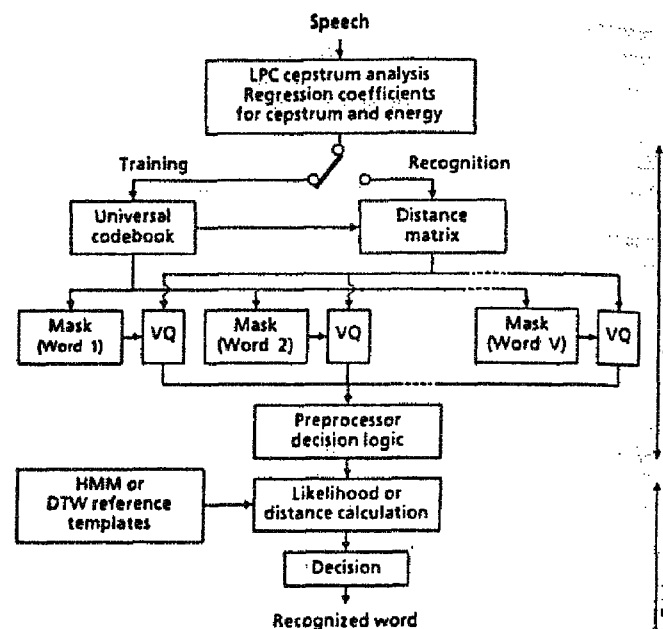


Fig. 3 - Block diagram of word recognizer incorporating a VQ preprocessor and an HMM- or DTW-based postprocessor.

combination of the transitional and the instantaneous LSP frequencies. We have also compared two ways of combining these two sets of parameters;

[Combination in the distance domain]

$$d_{LSP} + d_{\Delta LSP} = \sum_{i=1}^m (x_i - y_i)^2 + w \sum_{i=1}^m (\Delta x_i - \Delta y_i)^2 \quad (1)$$

where, x_i and y_i are the i -th LSP frequency of input speech and reference speech, respectively, and w is a weighting factor set to the inverse of the mean variance.

[Combination in the parameter domain]

$$\begin{aligned} d_{LSP + \Delta LSP} &= \sum_{i=1}^m \{(x_i + w\Delta x_i) - (y_i + w\Delta y_i)\}^2 \\ &= \sum_{i=1}^m \{(x_i - y_i) + w(\Delta x_i - \Delta y_i)\}^2 \end{aligned} \quad (2)$$

Experimental results show that the latter method has a slightly better recognition performance than the former method, and that the performance is better than that obtained using the cepstral and Δ cepstral coefficients. Combination in the parameter domain is also advantageous in terms of computation; that is, the

combination can be obtained during speech signal analysis and thus, it does not result in extra computation at the recognition stage [Gurgen et al., 1990].

2.3 On the Use of Prosodic Features in Speech Recognition

Speech wave conveys both spectral information and prosodic information. Prosodic information includes pitch, duration, and power, which are difficult to use in speech recognition. We have recently tried to use pitch information in both isolated word recognition and continuous speech recognition.

Figure 4 shows a block diagram of the word recognition system. The system has two major flows. The left flow is a phonetic recognition part based on HMMs. The other side is for pitch pattern recognition. Pitch pattern of input speech is compared with pitch pattern templates using a DTW technique. Pitch pattern templates are produced by averaging pitch patterns obtained from a set of words having the same accent pattern. A distance measure has been proposed based on a combination of the pitch pattern distance and the phonetic likelihood. Speaker-dependent word recognition experiments were carried out, and it was shown that the proposed measure reduces the recognition error rate from 1.62% to 0.88%, which is roughly a 45% reduction compared with the conventional phonetic likelihood measure [Takahashi et al., 1990].

This method has been extended to continuous speech recognition, in which pitch patterns of Japanese minimal phrases are modeled by HMMs and used for phrase boundary detection. The experimental results show that roughly 70% of the phrase boundaries can be correctly detected by this method [Takahashi et al., 1991].

III. SPEAKER ADAPTATION

3.1 Limits of Speaker-Independent Methods

A number of approaches have been tried in an effort to build speaker-independent recognition systems, typically under HMM-based frameworks. An HMM is a powerful and accurate stochastic model, which can represent speech variations when a sufficiently large amount of training speech is available.

One of the disadvantages of the speaker-independent approach is that it neglects various useful characteristics of the speaker in spite of the fact that they can be learned after recognition of several words or sentences [Furui, 1990b]. If these characteristics can be properly used, the recognition process is expected to be accelerated due to the narrowing of the search space. Another disadvantage is that when the distributions of feature parameters are very broad or multi-modal, such as when male and female voices as well as various dialects are combined, it is difficult to separate phonemes using speaker-independent methods. In these situations, nonsensical matches are probable. For example, the first and the second halves of a word may be matched against the male and female halves of a bimodal distribution, respectively. To cope with these problems, it is essential to introduce speaker-adaptation techniques.

Speaker adaptation or normalization ("speaker adaptation" will indicate both adaptation and normalization hereafter) is a method of automatically adapting reference templates to each new speaker or normalizing (reducing) interspeaker variations in each input speech, based on the transformation rules obtained using a few training words or short sentences. In large vocabulary recognition systems, training with respect to the utterances of all the vocabulary words is too troublesome for the user and consequently unrealistic. Training on a few words or short sentences is, therefore, a practical and realistic solution, although it is less convenient than the speaker-independent systems which have no need for training at all.

Speaker-adaptation methods are generally classified into supervised (text-dependent) methods in which training words or sentences are known, and unsupervised (text-independent)

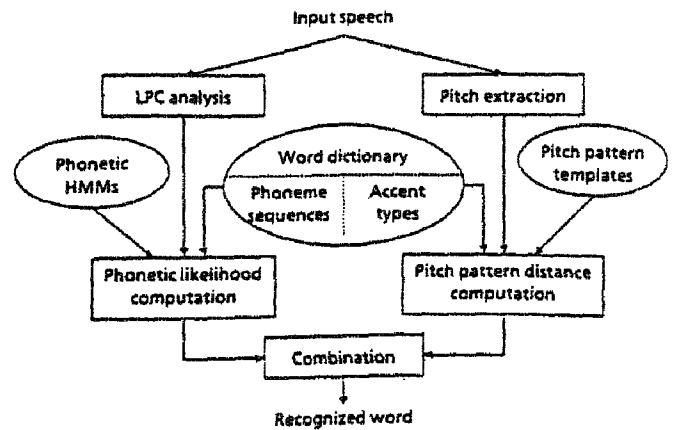


Fig. 4 - Block diagram of word recognizer using spectral and pitch pattern information.

methods in which arbitrary utterances can be used. Both methods can also be classified as offline methods in which training words or sentences must be uttered before the recognition, or as online methods in which utterances for recognition are, at the same time, used for training.

Ideally for users, the system should work as if it were a speaker-independent system which requests no additional training utterances from each speaker. The system should also adapt to the speaker's voice automatically using utterances for recognition. Such a system can be achieved by the unsupervised, online adaptation mechanism.

3.2 Codebook Adaptation/Normalization Method

Speaker-adaptive parameters can be estimated from speaker-independent parameters according to mapping rules. The mapping rules are estimated from the relationships between speaker-independent and speaker-dependent parameters. Within the framework of VQ-based speech recognition, both supervised and unsupervised methods of adapting the speaker-independent codebook to a new speaker or normalizing (adjusting) the input speech to the codebook have been proposed. Each word is represented by an HMM or single/multiple sequences of codebook entries in the word dictionary. Individual variations on how a word is uttered are modeled by the HMM or the multiple code sequences according to these methods. The HMM and code sequences are not changed during the adaptation and are universally used for all speakers.

3.2.1 Supervised Adaptation

For supervised adaptation, the mapping rules are obtained through DTW or a forward-backward algorithm. Figure 5 is a block diagram of a supervised adaptation method using DTW [Shikano et al., 1986]. The utterances of a reference speaker are used to create an initial codebook. These utterances are vector-quantized, that is, converted into sequences of codebook entries. In the training stage, training utterances of a new speaker are converted into code sequences and time-aligned with the same word or sentence uttered by the reference speaker, using the DTW technique. The spectral mapping function between the codebook elements of these two speakers is obtained from alignment functions, that is, the correspondences between the time axes.

Each codebook element is included in various words, and each codebook element of the reference speaker corresponds to various elements of the new speaker. Thus, a histogram of correspondences between codebook elements of the reference speaker and the new speaker, that is, a histogram of co-occurrences of codebook elements, is calculated using the alignment results of all training words or sentences. The mapping function is weighted by the

histogram to find the best correspondence rule.

In the recognition stage, input speech is vector-quantized and mapped (normalized) to the reference speaker's spectrum using the mapping rules at every frame. The similarity between the normalized input speech and each word of the reference speaker is then calculated and used in the recognition decision.

3.2.2 Unsupervised Adaptation

Figure 6 is a block diagram of an unsupervised codebook adaptation method [Furui, 1989b, Furui, 1989c]. The idea of this method is based on an adaptation algorithm for a segment vocoder [Shiraki et al., 1990]. First, an initial codebook and a VQ-indexed word dictionary are prepared. The codebook size was set as 1,024 in the experiment. The initial codebook is produced by clustering the voices of multiple speakers, and commonly serves as the initial condition for each new speaker.

In the adaptation process, a set of spectra from the training utterances of a new speaker and the reference codebook elements are clustered hierarchically in an increasing number of clusters. Using the deviation vectors between centroids of the training spectra clusters and the corresponding codebook clusters, either codebook elements or input frame spectra are shifted so that the corresponding centroids coincide. Continuity between adjacent clusters is maintained by determining the shifting vectors to be the weighted-sum of the deviation vectors of adjacent clusters. Adaptation is thus performed hierarchically from global to local individuality.

Figure 7 illustrates the hierarchical adaptation procedures for shifting the codebook elements from the beginning to the four-cluster stage. The size of the codebook (1,024) is maintained throughout the adaptation process. Using the hierarchical technique, phonetic correspondence between training utterance spectra and codebook elements is roughly maintained.

Several modifications to the adaptation method have also been investigated [Furui, 1989b].

3.3 HMM Adaptation

We recently proposed a new speaker-adaptive speech recognition method using a stochastic speaker classifier [Imamura, 1991].

In the conventional statistical speech recognition method, the main problem was to find the best word sequence W' which maximizes the *a posteriori* probability $P(W|Y)$ for input acoustic string Y . Using Bayes' rule, the *a posteriori* probability $P(W|Y)$ is computed as

$$P(W|Y) = P(W)P(Y|W)/P(Y) \quad (3)$$

where $P(Y)$ is the *a priori* probability of acoustic string Y , $P(W)$ is the *a priori* probability of word sequence W which is given by a language model, and $P(Y|W)$ is the category-conditional probability which is given by an acoustic model such as a sub-word or whole-word HMM.

In our method, the *a posteriori* probability is given by

$$P(W|Y, S) = P(W)P(Y, S|W)/P(Y, S) \quad (4)$$

where $P(Y, S)$ is the *a priori* joint probability of acoustic string Y and speaker individuality S , and $P(Y, S|W)$ is the category-conditional joint probability which is given by a speaker constrained acoustic model. The major problems are: "How to measure the speaker individuality S ?" and "How to compute the $P(Y, S|W)$?"

Figure 8 shows a block diagram of the speaker-adaptive recognition method. In this method, the stochastic speaker classifier is used as the feature extractor for speaker individuality information. The speaker classifier includes several (K) speaker classes represented by speaker Markov models. The speaker

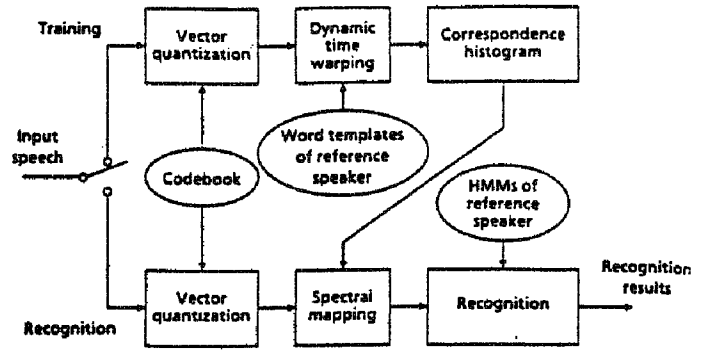


Fig. 5 - Block diagram of supervised speaker adaptation by spectral mapping.

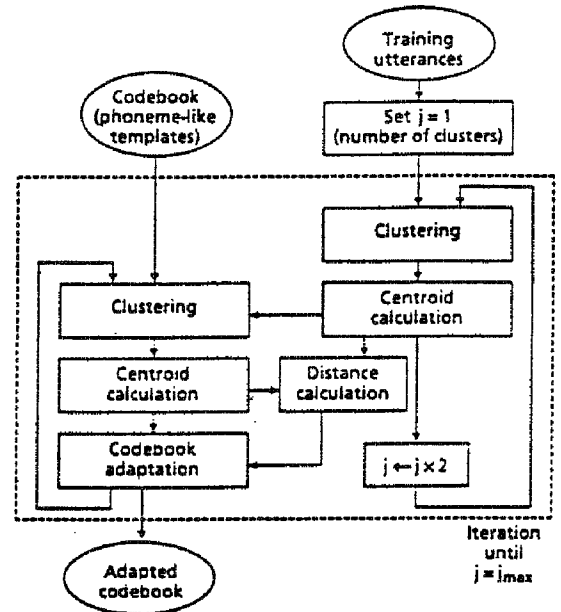


Fig. 6 - Block diagram of unsupervised codebook adaptation.

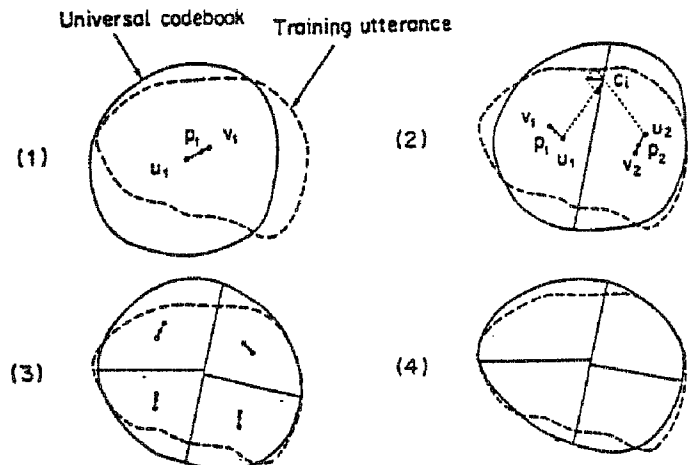
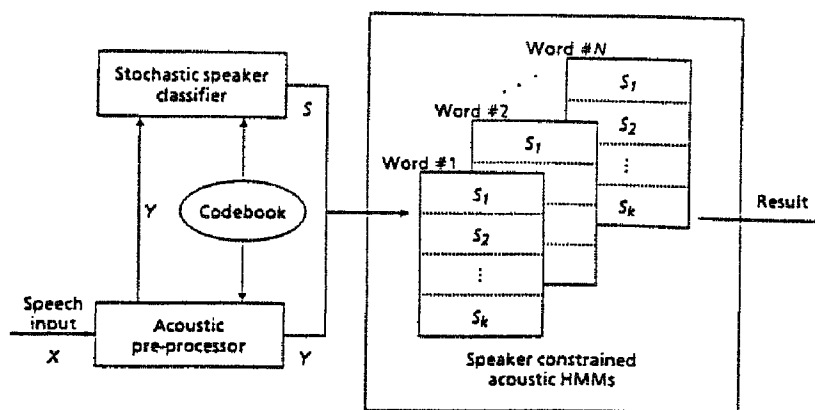


Fig. 7 - Hierarchical codebook adaptation algorithm maintaining continuity between adjacent clusters. v_j : centroid of j -th training spectra cluster, u_i : centroid of corresponding codebook cluster, p_j : deviation vector, and c_i : shifting vector for i -th codebook element.

Fig. 8 - Structure of a speaker-adaptive speech recognition method using a stochastic speaker classifier.



Markov models are estimated by clustering the training speech uttered by many speakers. For each input speech token Y , the classifier computes the category-conditional probability $P(Y|S)$ for each speaker class S . This probability indicates which speaker class (feature sub-space) is most suitable for the input speech. The obtained category-conditional probabilities $P(Y|S)$ are considered to be the quantization results of speaker individuality. In the subsequent word decoding phase, the speaker constrained acoustic HMMs which consider the output of the speaker classifier and the output of the acoustic pre-processor as input symbols are used to compute $P(Y, S|W)$.

Evaluation experiments were performed using a telephone speech database of 50 command words and 10 Japanese digits. Using four 9-state ergodic speaker HMMs estimated from the command words uttered by 116 training speakers, a word recognition accuracy of 98.1% was achieved for the 10 digits uttered by 116 different test speakers. The error rate is half of that obtained by the conventional speaker-independent (pooled training) method.

IV. PHONEME MODELS

4.1 Phoneme Units for Large-Vocabulary Recognition

In large-vocabulary recognition systems, it is impossible to store all words as spectral time sequences. Therefore, it is desirable to prepare phonetic units and to store each word as a sequence of these indices. Since phoneme spectra within continuous speech vary as a result of the influence of preceding and succeeding phonemes, we proposed a method of automatically creating context-dependent phoneme units by means of statistical analysis of a speech database [Sagayama, 1989]. Vocabulary words are represented by a tree-structured concatenation of these units.

4.2 Robustness of HMM Phoneme Models against Speaking Style Variations

For use in real situations, recognition systems must be able to cope with speech variations. The variations can be classified as those due to the speaker and those due to the surrounding environment. The former includes individuality, dialect, fluency, stress, speaking rate, level, pitch, and so on. Variations of phonetic features and transitions between them can be represented efficiently and flexibly in a probabilistic manner by HMMs, which are trained using utterances from many speakers. The HMM method has another advantage in that it uses relatively little computation for recognition, since each word or phoneme is represented by a sequence of a small number of states.

We have examined robustness of six types of phoneme-HMMs against speaking-style variations [Matsuoka et al., 1991]. The six types are VQ- and fuzzy VQ-based discrete HMMs, and single-Gaussian and mixture-Gaussian HMMs with either diagonal or full covariance matrices. Eighteen Japanese-consonant

recognition experiments were performed using isolated word utterances, phrase-by-phrase utterances, and fluently spoken sentence utterances. The mixture-Gaussian HMM with diagonal covariance matrices, the fuzzy VQ-based discrete HMM and the single-Gaussian HMM with full covariance matrices displayed better results than the other three types, when different speaking-style utterances were used in the training and testing.

We have also proposed a new model-adaptation technique that combines multiple models using the deleted interpolation method.

4.3 Fuzzy Training of Neural Net Phoneme Models

Generalization is one of the important issues in using neural networks (NNs) for speech recognition. From this point of view, we have proposed a fuzzy training algorithm, and applied it to phoneme recognition by one-hidden-layer tied-connection NN [Gurgen et al., 1991]. This algorithm is an alternative to the conventional mean square error (MSE) back propagation with hard-decision supervision. The conventional algorithm uses a hard decision criterion (1 and 0 supervisor signals) and suffers from the overlearning problem. In contrast, the proposed training algorithm uses fuzzy-decision supervision. In phoneme sample space, the proposed algorithm takes into account the overlap regions of phoneme boundaries.

The supervisor signal of each sample of training data is determined by the grade of membership using neighboring samples including the original sample. The grade of membership for each component F_k of the supervisor signal is

$$F_k = \sum_{j \in C_k} \exp(-d_j) / \sum_{j=1}^N \exp(-d_j) \quad (5)$$

where C_k is the k -th phoneme class, d is the Euclidean distance, and N is the number of nearest neighbors. The fuzzy effect of N -nearest neighbors on each training sample is computed and the NN is trained with these training samples using the MSE criterion. This method therefore softens the hard decision criterion and reduces the overlearning problem. Initiation of the fuzzy training is done with the conventional hard-decision back propagation method to achieve a good initial weight set.

The proposed algorithm provides better generalization for different speaking styles (isolated word, phrase, and continuous speech) than the conventional back propagation algorithm.

V. CONTINUOUS SPEECH RECOGNITION SYSTEMS

5.1 System Using Dependency Analysis

In our continuous speech recognition system (Fig. 9) [Matsunaga et al., 1990], input speech is represented by time

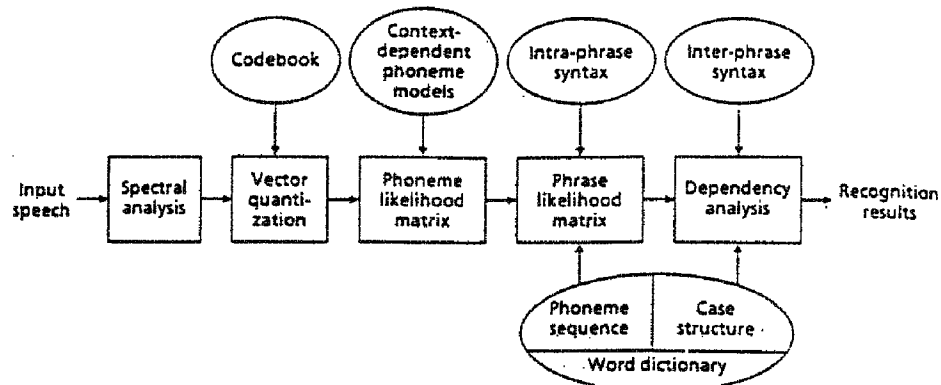


Fig.9 - Structure of continuous speech recognition system based on two-level grammar approach.

sequences of cepstral and Acepstral coefficients. The time sequences are then converted into code sequences using a vector quantization technique. The vector code sequences are compared with the context-dependent phoneme units and likelihoods for the occurrence of various phonemes are stored in a likelihood matrix. Then likelihoods for the occurrence of phrases are calculated using the phoneme likelihood matrix, a Japanese word dictionary, and a transition network grammar for constructing Japanese phrases.

Finally, the sentence is recognized as a sequence of most likely phrases by considering inter-phrase syntactic and semantic likelihoods calculated on the basis of an inter-phrase dependency analysis. A joint likelihood, combining acoustic, syntactic, and semantic likelihoods derived from acoustic processing and linguistic processing, is maximized to obtain the optimum solution. The procedure takes into account the redundancy of speech and the large freedom of phrase order which is a characteristic of the Japanese language.

The system was evaluated using a 71 sentence speech (418 phrases) uttered by two speakers, one male and one female. Two training sets were used: 216 phonetically balanced words, and a combination of 216 words and 29 sentences. The word dictionary has 360 entries and perplexity of the phrase syntax is 40. Certainty factors of dependency relationships were empirically determined through the analysis of technical literature. Table 1 shows the experimental results. The dependency parser increased the average phrase recognition rate from 69.2% to 83.1%, when the

Table 1 - Sentence speech recognition results

HMM training data		216 words		216 words + 29 sentences	
Dependency analysis		no	yes	no	yes
Phrase recognition rate (%)	male	45.5	55.3	63.6	78.9
	female	54.5	75.4	74.8	87.3
	average	50.6	65.3	69.2	83.1

training set of 216 words and 29 sentences was used. These results show the effectiveness of the semantic dependency analysis.

5.2 Japanese Dictation System for Medical Doctors

We are now developing a Japanese dictation system, voice-input word processor, for medical doctors using continuous speech recognition technology [Tsuboi et al., 1990]. The system is speaker-dependent and recognizes continuous phrasal speech.

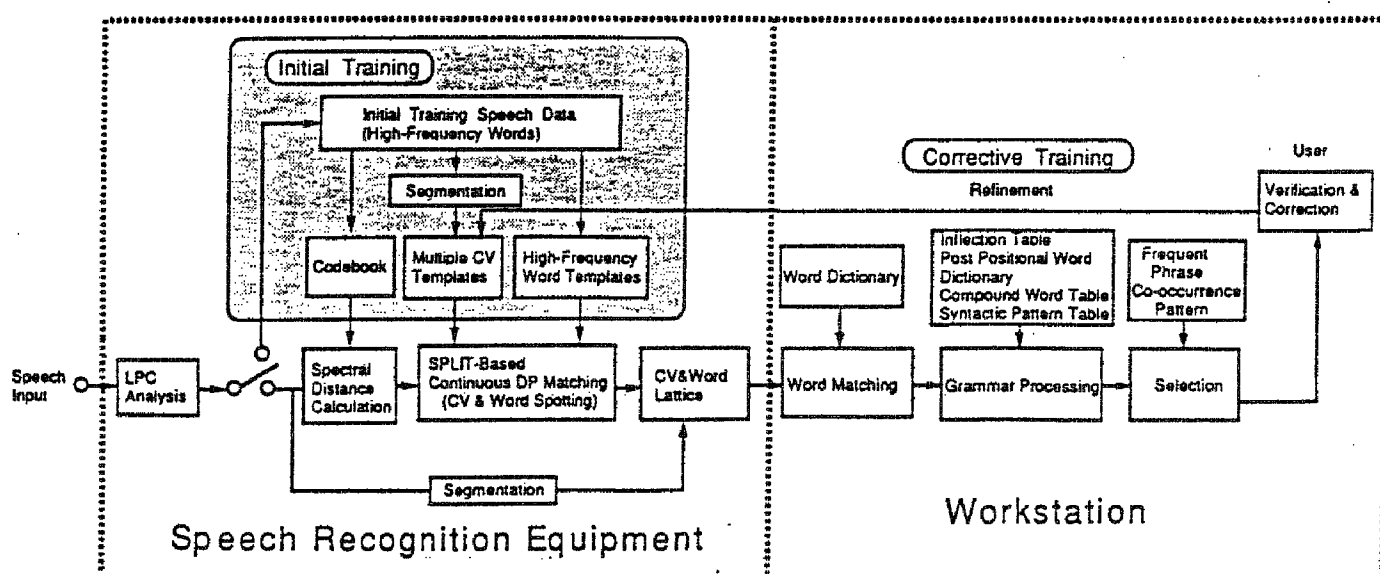


Fig. 10 - Japanese dictation system for medical doctors using continuous speech recognition.

Figure 10 illustrates the structure of the system, which consists of acoustic processing and linguistic processing parts. These two parts are implemented by a speech recognition equipment and a workstation, respectively. In the former part, a CV (consonant-vowel) syllable lattice and a word lattice are generated from input speech using a VQ-based continuous DP-matching technique. These lattices are converted into written Japanese form using a word dictionary and grammar information. The CV templates are automatically adapted to each speaker using the recognition results. Japanese conversion accuracies at this moment for X-ray CT scanning reports are roughly 80% for normal reports, and 65% for abnormal reports.

5.3 Dictation System Using Phoneme Source Modeling

We are investigating a phonetic typewriter that utilizes the underlying syntactic and statistical structure of phoneme and character sequences [Yamada et al., 1991]. A schematic diagram of the system, which consists of an HMM-based acoustic processing part and phoneme source modeling, is shown in Fig. 11. A syllable trigram approach to language source modeling is effective and promising for the Japanese language, since Kana (Japanese syllabary alphabet) roughly corresponds to consonant-vowel (CV) syllables.

For our phonetic typewriter, a general Japanese syllable sequence structure is written using context-free rewriting rules, and this structure is precompiled into the form of an LR table with syllable trigram probabilities. These syllable trigram probabilities are obtained by using a large text database. The predictive LR parser predicts possible phoneme sequences from left to right according to the general Japanese phoneme sequence syntax. The parser calculates phoneme sequence probabilities based on syllable trigram and HMM probabilities, and the system recognizes speech as the phoneme sequence with the highest probability.

The phonetic typewriter has been tested using 279 phrases uttered by one male speaker, and the syllable source model has achieved a 94.9% phoneme recognition rate with the test-set phoneme perplexity of 3.9. Without the syllable trigram, the phoneme recognition rate was only 73.2%.

A trigram model based on character (Kana and Chinese character) sequences in usual Japanese sentences has also been studied. It has been indicated that the character trigram model can significantly reduce the phoneme perplexity, compared with the syllable trigram model.

5.4 Word Spotting System

One of the important issues regarding man-machine interactive systems using speech recognizers is how to cope with noise and non-vocabulary words. As a solution to this problem, we have implemented an HMM-based speaker-independent word spotting system using Transputers [Imamura et al., 1990]. Candidates of word end-points and corresponding likelihood scores are computed with the continuous Viterbi decoding algorithm. To prune unreasonable candidates, three new methods have been introduced to duration control, threshold logic (for the likelihood scores), and local peak detection. An efficient parallel processing scheme for word spotting is carried out by using a tree structure of Transputers. In each frame period, the spectral feature vector from the speech analyzer is broadcasted from a root Transputer to eight node Transputers. With this structure, 72 words can be processed within a 12 ms frame period.

Word spotting experiments were conducted using 10 Japanese digits as keyword vocabulary. The system was trained using a database of keywords uttered isolatedly by 107 male speakers. A database of 100 short sentences, which were uttered by 10 new male speakers and recorded over noisy telephone networks, were used for testing. Each sentence included one keyword. Experimental results show that when the pruning is controlled to set the false alarm rate at 3%, a word detection accuracy of 97% can be obtained.

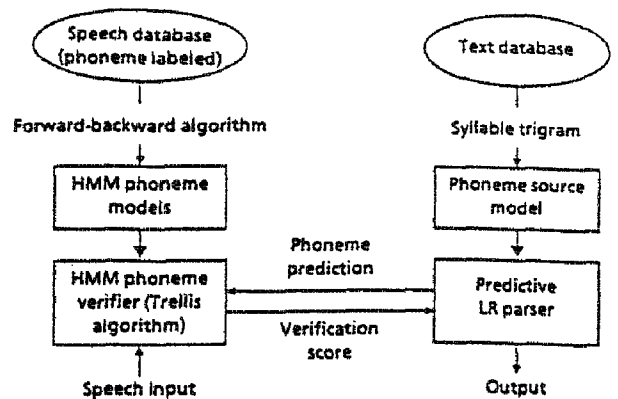


Fig. 11 - Schematic diagram of phonetic typewriter based on the HMM-LR method.

VI. SPEAKER IDENTIFICATION AND VERIFICATION

We have proposed a VQ-based text-independent speaker recognition method robust against text-to-text and inter-session variations [Matsui et al., 1991]. Figure 12 shows the principal structure of the system, incorporating three key techniques to cope with these spectral variations. First, either an ergodic HMM or a V/UV (voiced/unvoiced) decision is used to classify input speech into broad phonetic classes. The figure shows the case where HMM-based classification is used. Second, a new distance measure, Distortion-Intersection Measure (DIM), is introduced for calculating VQ distortion of input speech against speaker-dependent codebooks. DIM is characterized by selective matching using only a stable subset of test speech in VQ distortion calculation. Third, a new feature normalization method, Talker Variability Normalization (TVN), is introduced. TVN normalizes parameter variations taking both inter- and intra-speaker variability into consideration. TVN emphasizes feature parameters that have a relatively large inter-to-intra-speaker variation ratio.

The system was tested using utterances of nine speakers recorded at four sessions over three years. Speaker-dependent codebooks were made at every session, and the utterances recorded at different sessions were tested against these codebooks. When a 2-state ergodic HMM was used for broad phonetic classification, cepstral and Acepstral coefficients were used as feature parameters, and an average speaker identification accuracy of 98.5% was obtained. When V/UV decision was used for the classification, cepstral and Acepstral coefficients were extracted in unvoiced periods of speech, and pitch and Apitch frequencies were additionally extracted in voiced periods. In the latter case, a speaker identification accuracy of 99.0% was achieved.

VII. FUTURE PROBLEMS

Speech recognition technology is expected to play important roles in future communication and information services. To achieve these new services, it is important to consider the following questions from the viewpoint of human-machine interface. "What are the desirable forms of human-machine interface?" "What kinds of systems are really comfortable for users?" "What are the conditions necessary to achieve a system which is helpful for intelligent human activities?"

From a technological point of view, future problems include topics related to speech individuality, robust and proper statistical modeling and new technologies such as sophisticated neural networks. The first topic is how to extract, process and normalize speech individuality. The second involves the problem of how to

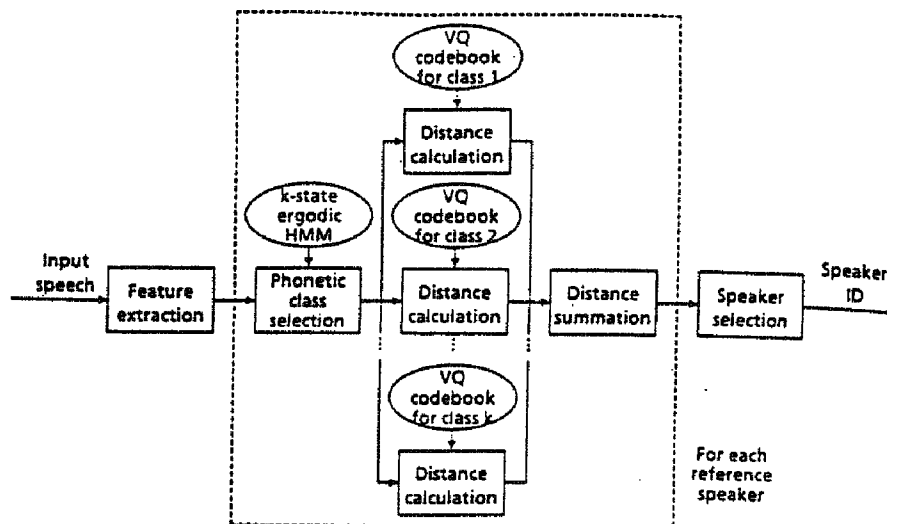


Fig. 12 - Block diagram of a text-independent speaker identification system using HMM-based broad phonetic classification.

produce robust statistical models based on restricted training data. Although HMM technology has brought speech recognition system performance to new high levels for a variety of applications, there still remain some fundamental problems. The third includes "How can we obtain advanced knowledge engineering technology, including knowledge acquisition and learning algorithms, which can handle human common sense?" and "How can we improve the neural network technology and combine it with conventional methods?"

ACKNOWLEDGMENTS

A number of people at NTT Laboratories have contributed to the works described in this paper. The author wishes to thank Dr. Kiyohiro Shikano, head of the Speech Information Processing Group, Dr. Hirokazu Sato, head of the Speech Processing Systems Group, and all staff members for their fruitful discussions and continuous research and development efforts.

REFERENCES

- Furui, S. (1986): "Speaker-independent isolated word recognition using dynamic features of speech spectrum," *IEEE Trans. Acoust., Speech, Signal Processing*, ASSP-34, pp.52-59
- Furui, S. (1989a): "Digital Speech Processing, Synthesis, and Recognition," Marcel Dekker, New York
- Furui, S. (1989b): "Unsupervised speaker adaptation method based on hierarchical spectral clustering," *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, Glasgow, Scotland, S6.9
- Furui, S. (1989c): "Unsupervised speaker adaptation based on hierarchical spectral clustering," *IEEE Trans. Acoust., Speech, Signal Processing*, ASSP-37, pp.1923-1930
- Furui, S. (1990a): "On the use of hierarchical spectral dynamics in speech recognition," *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, Albuquerque, S15a.10
- Furui, S. (1990b): "Speaker-dependent-feature extraction, recognition and processing techniques," *Proceedings of the ESCA Tutorial and Research Workshop on Speaker Characterization in Speech Technology*, pp. 10-27
- Gurgen, F., Sagayama, S. and Furui, S. (1990): "Line spectrum pair frequency-based distance measures for speech recognition," *Proc. Int. Conf. Spoken Language Processing*, Kobe, 13.1
- Gurgen, F., Aikawa, K. and Shikano, K. (1991): "The improvement of phoneme recognition performance of a neural network using fuzzy training," *Proc. SYNAPSE'91*, Osaka
- Imamura, A. and Suzuki, Y. (1990): "Speaker-independent word spotting and a transputer-based implementation," *Proc. Int. Conf. Spoken Language Processing*, Kobe, 13.5
- Imamura, A. (1991): "Speaker-adaptive HMM-based speech recognition with a stochastic speaker classifier," *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, Toronto, 58.S13.
- Itakura, F. (1975): "Line spectrum representation of linear predictor coefficients," *J. Acoust. Soc. Am.*, 57, S35(A)
- Mariani, J. (1989): "Recent advances in speech processing," *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, Glasgow Scotland, S9.1
- Matsui, T. and Furui, S. (1991): "A text-independent speaker recognition method robust against utterance variations" *Proc IEEE Int. Conf. Acoust., Speech, Signal Processing*, Toronto 57.S6.3
- Matsunaga, S., Sagayama, S., Homma, S. and Furui, S. (1990): "A continuous speech recognition system based on a two-level grammar approach," *Proc. IEEE Int. Conf. Acoust., Speech Signal Processing*, Albuquerque, S11.7
- Matsuoka, T. and Shikano, K. (1991): "Robust HMM phoneme modeling for different speaking styles," *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, Toronto, 10.S5.4
- Paliwal, K.K. (1990): "A study of LSF representation for speaker-dependent and speaker-independent HMM-based speech recognition systems," *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, Albuquerque, S15a.13
- Sagayama, S. (1989): "Phoneme environment clustering for speech recognition," *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, Glasgow, Scotland, S8.3
- Shikano, K., Lee, K.-F. and Reddy, R. (1986): "Speaker adaptation through vector quantization," *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, Tokyo, 49.5
- Shiraki, Y. and Honda, M. (1990): "Speaker adaptation algorithms based on piece-wise moving adaptive segment quantization method," *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, Albuquerque, S12.5
- Takahashi, S., Matsunaga, S. and Sagayama, S. (1990): "Isolated word recognition using pitch pattern information," *Proc. Int. Conf. Spoken Language Processing*, Kobe, 13.9
- Takahashi, S. and Matsunaga, S. and Shikano, K. (1991): "Accent phrase boundary detection in continuous speech using hidden Markov models," *Proc. Spring Meeting of Acoust. Soc. Jap.*, 2-5-13 (in Japanese)
- Tsuboi, T. and Sugamura, N. (1990): "A prototype for a speech-to-text transcription system," *Proc. Int. Conf. Spoken Language Processing*, Kobe, 20.8
- Yamada, T., Hanazawa, T., Kawabata, T., Matsunaga, S. and Shikano, K. (1991): "Phonetic typewriter based on phoneme modeling," *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, Toronto, 56.S3.4

A CONTINUOUS SPEECH RECOGNITION SYSTEM BASED ON A TWO-LEVEL GRAMMAR APPROACH

Shoichi Matsunaga, Shigeki Sagayama, Shigeru Homma and Sadaoki Furui

NTT Human Interface Laboratories
Midori-cho, Musashino-shi, Tokyo 180 Japan

ABSTRACT

This paper describes a Japanese continuous speech recognition system based on phonetic hidden Markov models (HMMs) combined with two levels of grammatical representations: an intra-phrase transition network grammar and an inter-phrase dependency grammar. A joint score, combining acoustic likelihood and linguistic certainty factor derived from phonetic HMMs and two levels of grammar, is maximized to obtain the optimal recognition results of sentences. Two efficient algorithms, bi-directional network parsing and breadth-first dependency parsing, are devised to globally optimize the joint score. The system attains a phrase recognition rate of 80.8% with the intra-phrase parser only, and 86.8% with both the intra-phrase and inter-phrase parsers, where the perplexity of the phrase syntax is 40. This result shows the effectiveness of the two-level grammar approach.

I. INTRODUCTION

Linguistic processing for speech recognition has been intensively studied from various approaches such as stochastic, syntactic and semantic grammars [1-3]. These grammars have advantages as well as limitations. Syntactic grammar is effective in describing the structure of phrases, while supposedly inadequate for describing the structure of whole sentences. On the other hand, case frame grammar [4] is suitable for governing the representation of the sentence structure. Thus, combining different types of grammars is practical for linguistic processing of speech recognition.

In Japanese sentences, which are sequences of minimal phrases, the phrase order is much less constrained than in English. On the other hand, the word order of phrases, which are short sequences of words, is very regular, and the sentence structure is ordered by semantic dependency between phrases. Syntactic constraints are useful in recognizing specific tasks or short-duration utterances. However, particularly in sentence recognition for phrase-order-free languages such as Japanese, semantic constraints are more powerful than sentence syntactic constraints.

We have developed a Japanese continuous speech recognition system which obtains the most likely sentence results taking account of acoustic, syntactic and semantic factors based on a two level grammar approach. This approach uses two

grammars which are an intra-phrase transition network grammar for phrase recognition and an inter-phrase dependency grammar for sentence recognition. The former is a syntactic grammar and the latter is a semantic and loose syntactic grammar. The dependency grammar is compatible with the case grammar, and has robustness against missing or misrecognized words.

Two efficient parsing algorithms are devised for each grammar. They are a bi-directional network parser and a breadth-first dependency parser.

The syntactic structure within phrases is represented by recursive transition networks (RTNs) to concisely cover a variety of phrase structures. With the network parser, input phrase utterances are parsed bi-directionally both left-to-right and right-to-left to reduce the amount of computation, and optimal Viterbi paths are found along which the accumulated phonetic likelihood is maximized.

With the dependency parser, inter-phrase dependency structures within a sentence are analyzed. Semantic certainty factor is determined taking into account grammatical cases incorporated in word dictionaries. The joint score, obtained by combining accumulated phonetic likelihood and semantic certainty factor derived from the dependency grammar, is maximized to obtain the optimal solution. The dependency parser utilizes efficient breadth-first search and beam search algorithms.

The approach described here is highly suitable for speech understanding systems since it can use semantic dependency structures. Furthermore, it is applicable to a wide range of tasks since it does not need any sentence syntax.

II. SPEECH RECOGNITION SYSTEM USING TWO LEVEL GRAMMAR

A block diagram of the system is shown in Figure 1. Input sentences are uttered phrase by phrase. After feature parameter extraction of the utterance, the parameter sequence is converted into a vector code sequence. Next, phonetic likelihood is calculated for every possible duration to obtain likelihood matrices for phoneme candidates, based on HMMs. Phonetic duration time can be easily controlled by giving maximum and minimum duration times to each phoneme. Next, phrase likelihood is calculated based on the phonetic likelihood matrices. The verb and adjective entries in the dictionary have

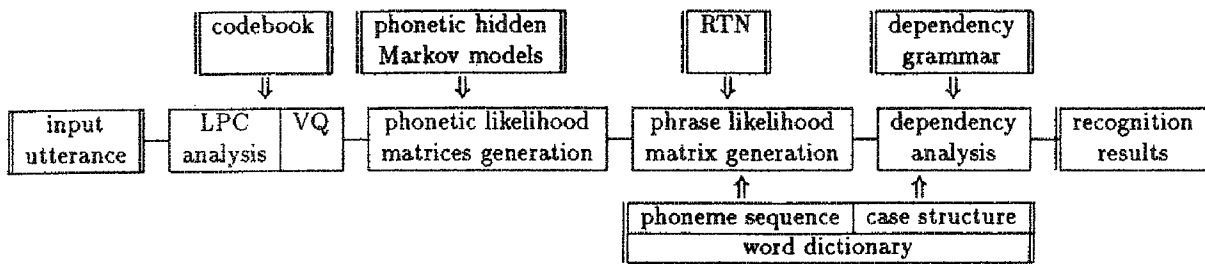


Figure 1: Block diagram of the continuous speech recognition system

grammatical cases, and the noun entries are accompanied by semantic primitives. Next, the top candidates of each likelihood are generated in a matrix form. The number of candidates is optional. Finally, using this matrix and the dependency grammar, the parser extracts the most likely sentence of a phrase sequence and its dependency structure.

III. INTRA-PHRASE SYNTACTIC GRAMMAR

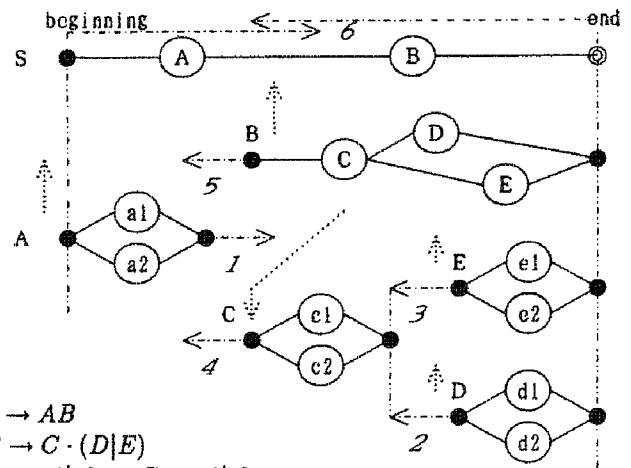
3.1 Syntactic Constraints of Phrases

Japanese phrases are composed of a stem part and a suffix part. The stem part is a verb, a noun, or an adjective. The suffix part is composed of suffixes such as auxiliary verbs and particles. Although connection of these words is very regular, there are many kinds of connection rules. Thus, to cover concisely this variety, the syntactic structure is represented by RTNs composed of sub-networks.

3.2 Speech Recognition of Phrases

When the phrase parser expands the RTNs to a single network in the phrase recognition process, the complexity of the phrase structure makes the network too large. Recognition is also time consuming. To overcome these problems, a bi-directional parser for RTNs is developed. This parser parses the input left-to-right in the stem part, and right-to-left in the suffix part. Bottom-up parsing is carried out for the stem words and the last suffixes of the suffix part, and top-down parsing is carried out for preceding suffixes. After calculating top-most m word candidates for each sub-network based on the Viterbi algorithm, the top-most m phrase candidates are generated.

A simple example of the phrase syntax and parsing flow is shown in Figure 2. S and B correspond to networks of the syntax, and A , C , D and E correspond to word sub-dictionaries. Processing flow is $A \rightarrow D \rightarrow E \rightarrow C \rightarrow B \rightarrow S$, and the processing order number is shown in the figure. First, the stem words, corresponding to A , are processed left-to-right from the beginning of speech. Next, the last suffixes, corresponding to D and E , are processed right-to-left from the end of speech. The calculation result of B is generated by combining the results of D , E and C . The likelihood calculation of C , namely the likelihoods of $c1$ and $c2$, is carried



$S \rightarrow AB$
 $B \rightarrow C \cdot (D|E)$
 $A \rightarrow a1|a2; \quad C \rightarrow c1|c2$
 $D \rightarrow d1|d2; \quad E \rightarrow e1|e2$

Figure 2: Example of phrase recognition process

out at every top-down scan of C . Finally, the phrase recognition results are generated by combining the results of A and B . In the figure, arrows \rightarrow , \leftarrow , \uparrow , \downarrow indicate the left-to-right process, the right-to-left process, the bottom-up control and the top-down control, respectively.

IV. INTER-PHRASE DEPENDENCY GRAMMAR

4.1 Semantic and Loose Syntactic Constraints

Dependency grammar is based on semantic dependency relationships between phrases. The syntactic rules satisfy the only two constraints. First, every phrase except the last must modify one and only one later phrase. This modification is called a dependency relationship or dependency structure. Second, no modification relationship between phrases in the sentence cross.

The semantic certainty factors of the dependency structure are easily provided using grammatical cases. There are two kinds of factors. One is associated only with dependency relationships of the modifier and modificant phrases: agreement between the semantic primitive of the modifier and that required by the modificant, agreement between the case of modifier and that required by the modificant, idiomatic expressions and so on. The other factor is associated with all

the dependency structures of the phrase sequence: a phrase with the obligatory case required by the modificant, no modification of the same phrase by different phrases having the same case, simplicity of the sentence structure and so on. The certainty factor values for these items are given heuristically.

4.2 Parser for Dependency Structure Grammar

This parsing is equivalent to solving the following objective function using the constraints of dependency structure grammar.

$$T = \max_{\{p\}} \left[\sum_{j=1}^N c(x_{j,p}) + \max_Y \sum_{j=1}^N \text{dep}(w_{1,j-1}, x_{j,p} | Y_{1,j,p}) \right] \quad (1)$$

where $1 \leq j \leq N$, $1 \leq p \leq M$, N is the number of input phrases, M is the maximum number of recognition candidates for each uttered phrase, $x_{j,p}$ is a candidate of the j -th input phrase with the p -th best likelihood, and $c(x_{j,p})$ is its log-likelihood. A phrase sequence with one phrase candidate for each i -th to j -th input phrase and whose last phrase is $x_{j,p}$ is denoted by $X_{i,j,p}$. $Y_{i,j,p}$ is one of the dependency structures of $X_{i,j,p}$. $w_{i,j-1}$ is the set of phrases that modify $x_{j,p}$ in the sequence $X_{i,j,p}$. Here, $\text{dep}(w, x | Y)$ is the linguistic certainty factor of dependency relationships between w and x taking Y into account. The first item of the term on the right in Eq.(1) is the summation of phonetic likelihoods of the hypothesized sentence composed of its phrase sequence, and the second item is the summation of linguistic certainty factor. Maximizing Eq.(1) gives the sentence and its dependency structure as the speech recognition result.

To solve Eq.(1) effectively, a fast parsing algorithm using breadth-first search and beam search was developed. This algorithm is based on the fundamental algorithms [5,6]. Although it offers sub-optimal solutions, it is practical because it requires much less processing than the depth-first search.

The breadth-first algorithm is formulated as follows. Its derivation is described in detail in reference [7]. First, $\text{dep}(w, x | Y)$ can be divided into two terms.

$$\text{dep}(w_{1,j-1}, x_{j,p} | Y_{1,j,p}) = \sum_{x \in w_{1,j-1}} \text{dep1}(x, x_{j,p}) + \text{dep2}(Y_{1,j,p}, x_{j,p}) \quad (2)$$

where dep1 is the certainty factor associated with dependency relationships of only the modifier and modificant phrases, and dep2 is the certainty factor associated with $Y_{1,j,p}$. Using notation $S(1, x_{j,p})$, the objective function's value of a phrase sequence including the top phrase to $x_{j,p}$ in the sentence, and $D(i, x_{j,p})$, the value of a phrase sequence not including the top phrase ($i \neq 1$), the recursive relation using beam search are derived.

$$S(1, x_{j,p}, r) = r^{th} \max_{k,q,r1,r2} [S(1, x_{k,q}, r1) + D(k+1, x_{j,p}, r2) + \text{dep1}(x_{k,q}, x_{j,p}) + \text{dep2}(Y_{1,j,p}, x_{j,p})] \quad (3)$$

$$D(i, x_{j,p}, r) = r^{th} \max_{k,q,r1,r2} [S(i, x_{k,q}, r1) + D(k+1, x_{j,p}, r2) + \text{dep1}(x_{k,q}, x_{j,p}) + \text{dep2}(Y_{i,j,p}, x_{j,p})] \quad (i \neq 1) \quad (4)$$

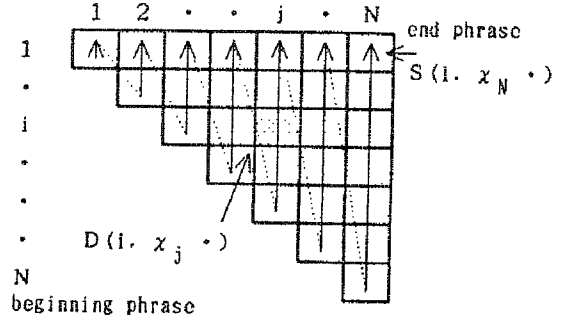


Figure 3: Configuration of parsing table

where $i \leq k \leq j-1$, $1 \leq q \leq M$, and $1 \leq r, r1, r2 \leq L$. Here, $r, r1$ and $r2$ indicate the beam ranks, L is the maximum number of beams, $S(1, x_{j,p}, r)$ and $D(i, x_{j,p}, r)$ are the r -th value of the element whose phrase sequence is $X_{i,j,p}$ and the dependency structure is $Y_{i,j,p}$. Here, $r^{th} \max[\cdot]$ is a function for deriving the r -th best value. When Eq.(3) or (4) is calculated, $Y_{i,j,p}$ is stored for use in the later stage of evaluating dep2 .

Initial values are given as follows.

if $i = 1$ (top phrase)

$$S(1, x_{1,p}, 1) = c(x_{1,p}) + \text{dep2}(Y_{1,1,p}, x_{1,p}) \quad (5)$$

if $i \neq 1$ (not top phrase)

$$D(i, x_{i,p}, 1) = c(x_{i,p}) \quad (6)$$

After calculating the recurrence relation, the value of the objective function is obtained:

$$T = \max_p [S(1, x_{N,p}, 1)] \quad (7)$$

where $1 \leq p \leq M$. The best sentence and its dependency structure are given through $Y_{1,N,p}$ where p maximizes Eq.(7). The parsing table is shown in Figure 3 and the parsing algorithm is shown in Table 1. In Figure 3, the first row corresponds to S , and others correspond to D . The phrase sequence for the first to N -th phrase corresponds to the rightmost top cell. Each cell is composed of M sub-cells for the number of candidates, and each sub-cell is composed of L sub-cells for the beam width. The arrows show the sequence of calculating the recurrence relation. The processing amount order for this algorithm is $O(N^3 M^2 L^2)$.

V. SPEECH RECOGNITION EXPERIMENTS

An input utterance was sampled at a rate of 12 kHz. One frame was extracted every 10ms with a 30ms Hamming

Table 1: Parsing algorithm

{1} : Loop for the end of phrase of the partial sequence DO {2} to {5} for $j = 1, 2, \dots, N$
{2} : Loop for the candidates DO {3} to {5} for $p = 1, 2, \dots, M_j$
{3} : Setting the initial value SET $S(1, x_{1,p}, 1)$ or $D(j, x_{j,p}, 1)$ (Eqs.(5),(6)) if $j = 1$, go back to {2}.
{4} : Loop for the beginning phrase of the partial sequence DO {5} for $i = j - 1, j - 2, \dots, 1$
{5} : Computation of recurrence relation { Loop for the end phrase of the former sequence } {5-1} : DO {5-2} to {5-4} for $k = i, i + 1, \dots, j - 1$ {5-2} : DO {5-3} to {5-4} for $q = 1, 2, \dots, M_k$ { Loop for the beam width } {5-3} : DO {5-4} for $r1 = 1, 2, \dots, L$ {5-4} : for $r2 = 1, 2, \dots, L$ * Evaluation of $S(1, x_{1,p}, r)$ or $D(j, x_{j,p}, r)$ taking account of $Y_{1,j,p}$ or $Y_{i,k,q}$ (Eqs.(3),(4)) * Store of $Y_{i,j,p}$
{6} : Acquisition of the parsing results * Detection of value p maximizing Eq.(7) * Acquisition of the phrase sequence and its dependency structure using $Y_{1,N,p}$

window and converted into 34 acoustic feature parameters: power, 16 LPC cepstra, Δ power and 16 Δ LPC cepstra [8,9].

In the training process, 216 phonetically balanced words were used. These utterances were manually labeled using 25 phoneme symbols including silence. Each phoneme was modeled by HMM and had 4 states and 7 transition paths. The parameter sequence was converted into a vector code sequence and a vector codebook composed of 256 prototype vectors was generated. Training of each HMM was carried out using the forward-backward algorithm, and the code sequence for training was cut out based on the phonetic labels. Output probabilities were floored after training.

In the testing process, input sentences were uttered phrase by phrase. Acoustic feature parameters of the input were generated in the same manner. These parameters were converted into a code sequence using the same speaker's codebook generated in the training process.

First, talker-dependent preliminary recognition tests were done on two sets of 216 words, one for training and the other for test, uttered by 10 speakers, 5 males and 5 females. The system attained a word recognition rate of 99.9% for the training set and 98.4% for the test set.

Next, talker-dependent recognition tests were performed on 100 sentences (including 668 phrases in an essay) uttered by the same speakers. The word dictionary had 360 entries and the perplexity of the phrase syntax was 40. The phrase syntax was capable of generating about 10^4 hypotheses for each phrase of speech. The number of candidates, M , was 5, and the beam width, L , was 8. Certainty factor values were empirically determined through feature-based speech recognition [7] of technical literature.

The system attained a phrase recognition rate of 80.8% using the intra-phrase syntactic parser only. The dependency

Table 2: Speech recognition results

data	216 words		668 phrases	
	training		test	
dependency analysis		-	without	with
recognition rate	99.9	98.4	80.8	86.8
(top 5)	100	100	96.8	-

parser increased this rate to 86.8%, as shown in Table 2. This result shows the effectiveness of the two-level grammar approach.

VI. CONCLUSION

This paper described a Japanese continuous speech recognition system using an intra-phrase transition network grammar and an inter-phrase dependency grammar. Input utterances were recognized efficiently to determine the best sentence using a bi-directional network parser and a breadth-first dependency parser. Recognition experiment results showed the effectiveness of the inter-phrase dependency grammar. The parser for this grammar can be easily expanded for sentence speech recognition [7].

Further development is currently in progress to refine phonetic models based on continuous HMMs which take context dependency into account.

ACKNOWLEDGMENT

The authors wish to express their appreciation to Hirokazu Sato of NTT Labs and Masaki Kohda of the University of Yamagata for their invaluable discussions. The authors also thank Frank Soong and Fred Juang of AT&T Bell labs for their useful suggestions.

REFERENCES

- [1] V.R.Leser, et al "Organization of the Hearsay II speech understanding system", *IEEE Trans. Acoust., Speech, Signal Processing*, ASSP-23 No 1, pp.11-24, 1975
- [2] H.Ney, "Dynamic programming speech recognition using a context-free grammar", *Proc. ICASSP*, pp.69-72, 1987, Dallas
- [3] K-F.Lee and H-W. Hon, "Large-vocabulary speaker-independent continuous speech recognition using HMM", *Proc. ICASSP*, pp.123-126, 1988, New York
- [4] S.Filmore, "The case for case." in Bach and Harms(eds.), 1-88, 1968
- [5] S.Matsunaga and M.Kohda, "Post-processing using dependency structure of inter-phrases for speech recognition." 1-1-23, *Proc. ASJ annual meeting*, pp.45-46, Mar.1986 (in Japanese)
- [6] K.Ozeki, "A multi stage decision algorithm for optimum bunsetsu sequence selection." *Paper Tec. Group, IECE Japan*, SP86-32, pp.41-48, Jul.1986 (in Japanese)
- [7] S.Matsunaga and M.Kohda, "Linguistic processing using a dependency structure grammar for speech recognition and understanding.", *Proc. COLING*, pp.402-407, 1988, Budapest
- [8] S.Furui, "Speaker-independent isolated word recognition using dynamic features of speech spectrum", *IEEE Trans. Acoust., Speech, Signal Processing*, ASSP-34 No 1, pp.52-59, 1986
- [9] S.Sagayama and F.Itakura, "On individuality in a dynamic measure of speech." 3-2-7, *Proc. ASJ annual meeting*, pp.589-590, Jul.1979 (in Japanese)

階層的スペクトル動特性を表現した符号帳による音声認識

正 員 古井 貞熙[†]

Speech Recognition Using VQ-Codebooks Representing Hierarchical Spectral Dynamics

Sadaoki FURUI[†], Member

あらまし 音声の各時点ごとのスペクトル特徴を、ケプストラムとパワーの静的特徴および階層的な動的特徴の組合せによって表現し、各単語あるいは各音素に含まれるこれらの特徴のクラスタ化によって作成した符号帳により、単語あるいは音素の特徴を表現する音声認識法を提案する。階層的な動的特徴は、各時点を中心とする長短の異なる長さの時間窓におけるパラメータ時系列の回帰係数として抽出する。こうして作成した各単語および各音素に特有の符号帳によって、入力音声をベクトル量子化し、音声区間における平均量子化ひずみの大きさによって、単語の候補を予備選択したり、単語や音素を認識したりすることができる。不特定話者の日本語 100 単語音声の認識実験の結果、本方法によって極めて効率良く候補単語を予備選択でき、高い精度で単語の認識ができることが確認された。また大語い単語中の /b//d//g/ の音素の認識実験の結果、従来の HMM、ニューラルネット等による方法と同程度の高い音素認識性能が得られた。本論文では更に、異なる種類の特徴を単一符号帳で表現する方法と、別々の複数符号帳で表現する方法の比較についても検討を行った。

1. ま え が き

音声知覚において重要な役割を果たしているスペクトルの動的特徴⁽¹⁾を、音声から適切に抽出し、音声認識に用いる方法を開拓することは、音声認識における重要な課題の一つである。このような方法の一つである、いわゆる Δ (デルタ) ケプストラム法 (ケプストラムや対数パワーの線形回帰係数の時系列を用いる方法)⁽²⁾は、音声認識において極めて有効であることが種々の実験で実証され、現在広く種々の音声認識系で用いられている^{(2),(3)}。更に、大語い単語音声認識において、ケプストラム、 Δ ケプストラムおよび Δ パワーを組み合わせたベクトルを要素とする符号帳を単語ごとに作成し、これらの符号帳でベクトル量子化 (VQ) したときのひずみを尺度とすることにより、候補単語が効率良く予備選択できることも、既に確認されている^{(4),(5)}。

これらの方法で、回帰係数を計算するフレーム数 (時間長) は、その設定値が認識性能に大きな影響を与えることはないので、通常、比較的良い結果をもたらす範囲の値 (40~100 ms) に適当に決めることが多い。

しかし、音声に含まれる重要な動的特徴は、実際には音素長、音節長などに対応して、種々の時間的単位の中に含まれていると考えられる。人間の聴覚においても、種々の時間分解能が存在することが示唆されている⁽⁶⁾。ここでは、音声の各時点ごとに、その時点を中心とする長短の異なる時間長について求めた Δ ケプストラムおよび Δ パワーを階層的に組み合わせて、これらのパラメータを要素とするベクトルの符号帳を、単語あるいは音素別に作成し、これによる VQ ひずみに基づく音声認識を試みた結果について述べる⁽⁷⁾。

異種の特徴ベクトルを組み合わせる場合、すべての特徴ベクトルを一括したベクトルについて符号帳 (単一符号帳) を作成する方法と、符号帳サイズをあまり大きくせずに VQ ひずみを小さく保つために、特徴ベクトルの種別ごとに符号帳 (複数符号帳) を作成し、個々の符号帳によるひずみを加え合わせて全体のひずみを計算する方法が用いられる。特に学習サンプルの数が少ない場合に、後者の方法を用いることが多いが、この方法は異なる特徴ベクトル間の独立性を前提にしているため、これらの間の相関が無視されるという問題がある。ここでは、単一符号帳と複数符号帳の効果の比較についても検討する。

2. では、不特定話者の単語音声認識において、VQ ひ

[†] NTT ヒューマンインタフェース研究所、武蔵野市
NTT Human Interface Laboratories, Musashino-shi, 180 Japan

ずみに基づいて候補単語を効率良く予備選択する方法とその効果について、3. では、VQ ひずみに基づいて、特定話者の大語い単語中の破裂子音/b//d//g/の認識をする実験とその結果について述べる。最後に4. で結論と今後の課題について述べる。

2. 単語候補の予備選択

2.1 方法

候補単語の予備選択機能を含む単語音声認識系全体の構成を図1に示す。主な実験条件は次のとおりである。

- (a) 認識対象：日本語 100 都市名
- (b) 学習サンプル：男性 4 名 (400 サンプル)
- (c) テストサンプル：上と異なる男性 20 名 (2,000 サンプル)
- (d) 標本化周波数：8 kHz
- (e) フレーム周期：8 ms, フレーム長：32 ms (ハミング窓)
- (f) 分析法：10 次 (LPC-) ケプストラム, 10 次 Δ ケプストラム, Δ パワー

学習用音声の話者 4 名は、30 名の男性話者のスペクトルの分析に基づき、その分布を代表する話者を選ん

だものである。文献(4), (5)では、 Δ ケプストラムと Δ パワーを求めるフレーム数を 7 フレームとしたが、今回は図2に示すように、階層的動特性として、7 フレーム (56 ms) とその 3 倍の 21 フレーム (168 ms) の 2 種類を組み合わせて用いる効果について検討した。各時点での音声特徴は、ケプストラムと合わせて、 $10(c) + 10(\Delta c_7) + 10(\Delta c_{21}) + 1(\Delta p_7) + 1(\Delta p_{21}) = 32$ 次元のベクトルで表現されることになる。ここで c はケプストラム, p は対数パワー, 引数は回帰係数を計算するフレーム数を示す。

各単語を特徴づける符号帳サイズはそれぞれ 64 とし、全単語の符号帳を総合した共通符号帳のサイズは、256, 512, 1,024 の 3 種類に変えた。具体的手順として

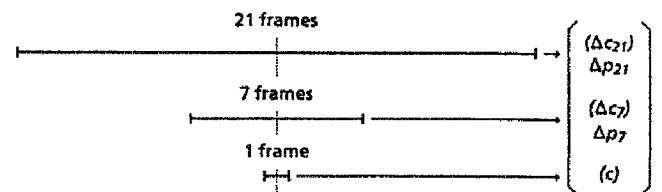


図2 階層的スペクトル動特性を表現する特徴ベクトルの構成

Fig.2 Structure of a feature vector representing hierarchical spectral dynamics.

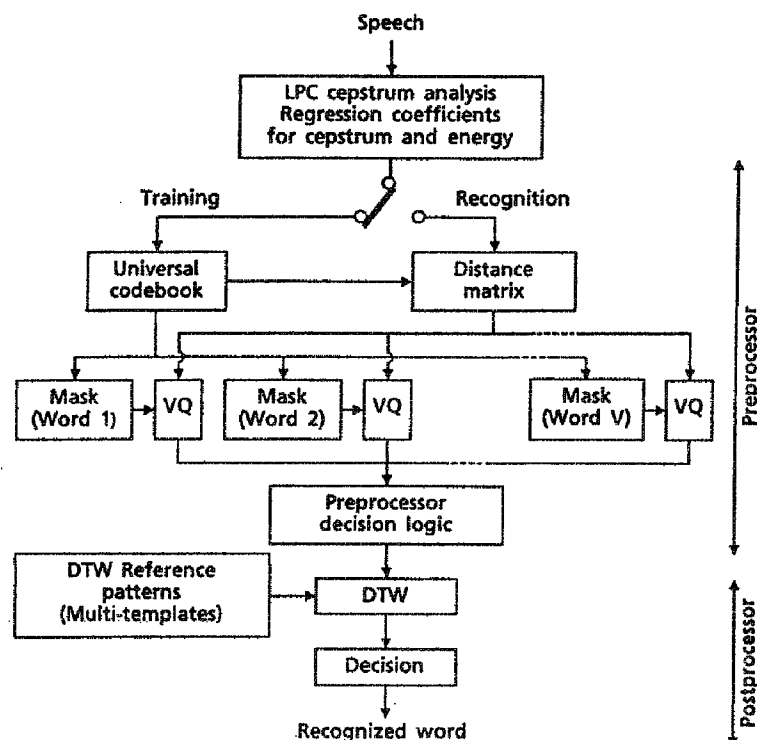


図1 予備選択と SPLIT 法の組合せによる単語認識系の構成

Fig.1 Block diagram of word recognizer incorporating a VQ preprocessor and a SPLIT post-processor.

表1 100単語音声の認識誤り率

方法	符号帳	符号帳サイズ		
		256	512	1024
VQ ひずみ*	単一符号帳($c, \Delta c_7, \Delta p_7$)	—%	13.45%	10.90%
	単一符号帳($c, \Delta c_{21}, \Delta p_{21}$)	—	16.45	10.40
	複数符号帳	—	9.05	6.65
	$D(c, \Delta c_7, \Delta p_7) + \alpha D(c, \Delta c_{21}, \Delta p_{21})$	—	9.05	6.65
	単一符号帳($c, \Delta c_7, \Delta p_7, \Delta c_{21}, \Delta p_{21}$)	—	4.45	3.60
SPLIT	単一符号帳(c)	6.50	6.20	6.50
	複数符号帳 $D(c) + \alpha D(\Delta c_7, \Delta p_7)$	2.15	2.35	2.25
	単一符号帳($c, \Delta c_7, \Delta p_7$)	3.00	2.15	1.95

* 各単語の符号帳サイズは64

は、まず各単語ごとに、4名の学習用音声の全フレームを LBG アルゴリズム⁽⁶⁾でクラスタ化して、サイズが64の符号帳を作成する。次に全単語の符号帳を組み合わせた集合($64 \times 100 = 6,400$ 要素)に再度 LBG アルゴリズムを適用して、上記の3種類のサイズの共通符号帳を作成する。最後に、各単語の符号帳要素を、最も距離が近い共通符号帳要素に置き換える。従って各単語の符号帳は、共通符号帳の部分集合になっている。

入力音声は、パラメータ系列に変換された後、各フレームごとに共通符号帳の全要素との距離が計算され、その結果は距離行列に蓄えられる。次にこの距離行列を用いて、入力音声を各単語の符号帳でベクトル量子化したときの量子化ひずみが計算される。各単語の符号帳は、共通符号帳の部分集合であるので、距離行列のうち、各単語の符号帳要素に対応する部分のみをマスクで取り出して、その範囲内での最小値を全フレームで加算することにより、容易に各単語符号帳に対する量子化ひずみを計算することができる。このひずみの値が、あらかじめ定めてあるしきい値よりも小さい場合は、その単語を正解の候補として残し(予備選択)、それ以外は候補ではないとして棄却する。選択された候補単語の中から、SPLIT 法によってどの単語であるかを決定する。SPLIT 法では、各単語が符号帳要素の時系列(4名の学習用音声から作成した4種類のマルチテンプレート)で表現されているので、上述の距離行列の値をそのまま用いて、容易に単語の判定を行うことができる。この予備選択と SPLIT 法の組合せにより、大語いの単語音声認識における演算処理量を大幅に削減することができる^{(4),(5)}。

距離(ひずみ)値を計算するときの各特徴量に対する重みは、次のように決めた。ケプストラムおよび Δ ケプストラムに関しては、それぞれ、各次数ごとの分散を全次数について平均し、全次数について一様に平均分散の逆数で重みづけた。 Δ パワーに対する重みは、従来の実験で経験的に決めてある定数(3.0)に分散の逆数を乗じた値に設定した。

2.2 認識結果

SPLIT 法の部分は用いずに、量子化ひずみに基づく予備選択の部分のみを用いて、ひずみの最も小さい単語に認識したときの認識誤り率を、SPLIT 法のみで認識した場合の結果と比較して表1に示す。表の中で()は、用いた特徴ベクトルすなわち符号帳の構成を示す。量子化ひずみによる実験に関しては、表の上から、動特性を抽出するフレーム数が7フレームの場合

($c, \Delta c_7, \Delta p_7$), 21フレームの場合($c, \Delta c_{21}, \Delta p_{21}$), 両者のひずみを組み合わせて重み付き加算した値を用いた場合、両者の動特性を単一のベクトル($c, \Delta c_7, \Delta p_7, \Delta c_{21}, \Delta p_{21}$)に組み合わせ、単一の符号帳にしてひずみを計算した場合、のそれぞれの結果を示した。いずれの場合も、静的な特徴であるケプストラムは、常に動特性と組み合わせて用いているので、特徴ベクトルの次元数はそれぞれ、21, 21, $21+21=42$, 32である。複数符号帳によるひずみの重み付き加算は、次のようにして行った(α :重み)。

$$D_{\text{sum}} = (D(c, \Delta c_7, \Delta p_7) + \alpha D(c, \Delta c_{21}, \Delta p_{21})) / (1 + \alpha) \quad (1)$$

SPLIT 法による実験に関しては、ケプストラムのみの場合、その符号帳と動的特徴(Δ ケプストラムおよび Δ パワー)の符号帳を別々(複数符号帳)に作成し両者のひずみを重み付き加算した場合、およびこれらの特徴を単一符号帳にした場合の結果を示した。

(1) 量子化ひずみによる方法の結果

表1の上半分の結果から、量子化ひずみによる認識方法において、7フレーム、21フレームそれぞれ単独の場合に比べて、両者のひずみ量を組み合わせるか、単一のベクトルに組み合わせてひずみを計算することによって、大幅に誤りが低下することがわかる。例えば、共通符号帳サイズが1,024の場合、7フレームまたは21フレーム単独符号帳による誤り率は、それぞれ10.90%と10.40%であるが、両者のひずみを加算した値による誤り率は6.65%に低下する。このことは、7フレームまたは21フレームから求めた動的特徴の認識における効果はほぼ同程度であるが、実際に抽出されている特徴には相違があり、このために組合せ効果

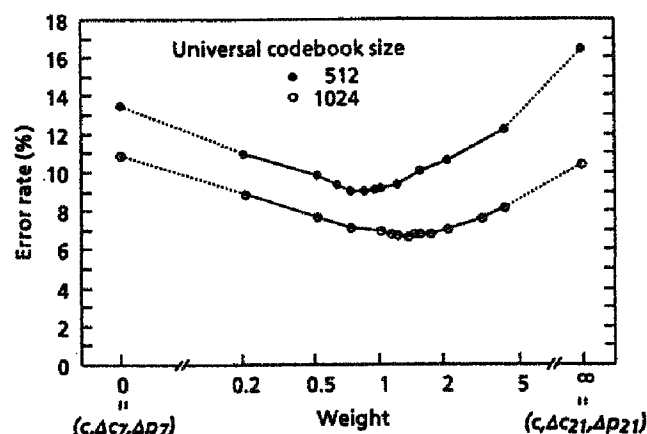


図3 7フレームと21フレームの回帰係数を含む2種類の符号帳によるVQひずみの組合せによる単語認識結果

Fig. 3 Error rates for word recognition based on the weighted sum of the two VQ-distortions obtained using the 7-frame and 21-frame regression coefficient codebooks, respectively.

が発揮されることを示している。

この両者のひずみを重み付き加算する場合の、重み α に関する認識誤り率の変化を図3に示す。重みの変化に対する組合せ効果の変動は小さく、重みの値はほぼ $\alpha=1$ 程度に設定すればよいことがわかる。

更に、表1の結果から、距離値で組み合わせるよりも、階層的動特性を含む単独の符号帳にした場合の方が、誤りが前者の1/2近くに低下し、共通符号帳サイズを1,024とすれば、3.60%の誤り率が得られることがわかる。

量子化ひずみによる認識方法における共通符号帳のサイズに関しては、一貫して1,024の場合の結果が512の場合の結果を上回っている。このため追加実験として、符号帳サイズを更に大きくした場合と、その極限（無限大サイズ）として共通符号帳をもたずに各単語の学習音声から作成したものをそのまま用いる場合について実験を行った。その結果、1,024よりも大きくしても、認識性能の向上は見られず、1,024が最適であることが確認された。

(2) SPLIT法による結果

表1の下半分の結果から、SPLIT法における共通符号帳のサイズに関しては、ケプストラムの単独符号帳、およびケプストラムと動的特徴の複数符号帳の場合は、256でほぼ十分であるが、ケプストラムと動的特徴を組み合わせた単一符号帳の場合は、512では不十分であり、1,024が必要であることがわかる。動的特徴を用いる場合の、複数符号帳と単一符号帳の結果を比較

すると、符号帳サイズが256の場合は複数符号帳、512の場合はほぼ同程度、1,024の場合は単一符号帳にした方が誤りが少なくなることがわかる。符号帳サイズが1,024の場合の、単一符号帳による誤り率は1.95%、複数符号帳による誤り率は2.25%である。

Kai-Fu LeeらのHMMを用いた音声認識実験⁽⁹⁾では、256のサイズの符号帳を用いて、単一符号帳よりも複数符号帳の方が良いことを示しているが、この結果はここで得られた実験結果とも一致している。符号帳サイズを大きくすることができれば、単一符号帳の方が良くなると予想される。

なお表1の最下段の、動的特徴を含む単一符号帳を用いる場合について、符号帳サイズに関する補足実験を行った結果、サイズは1,024で十分であり、ベクトル量子化を行わなかった場合の誤り率は、1,024の場合の誤り率（1.95%）にほぼ等しいことが確かめられた。

(3) 量子化ひずみによる方法とSPLIT法の比較

以上の実験から、7フレーム、21フレームの2種類の動特性を組み合わせた符号帳を用いれば、DPやHMMを用いなくても、不特定話者の100単語音声の認識において、3.6%の認識誤り率が得られることがわかった。この値は、動的特徴（ Δ ケプストラムおよび Δ パワー）を用いないSPLIT法の場合の約1/2である。 Δ ケプストラムおよび Δ パワーで表現される動特性は、音声の比較的ミクロな時間特性であり、DPやHMMで表現される比較的マクロな時間特性とは異なるが、上の実験結果は、ミクロな時間特性の集積でも単語の特徴をとらえることができることを示している。

なお、SPLIT法の場合に7フレームと21フレームの動特性を組み合わせる効果に関する実験も行ったが、この場合はほとんど効果が認められなかったため、この結果は省略した。より複雑なタスクで実験を行えば効果が得られると思われる。

2.3 予備選択の結果

次に、第1位のみでなく、距離値にしきい値を設けて、しきい値より小さい値をもつ複数の候補を予備選択する場合について検討した。しきい値(θ)は、式(2)に示すように、各入力に対して全候補との距離(D_n)の内の最小値を求め、その値に一定のバイアス(θ_0)を加えた値に設定した。

$$\theta = \theta_0 + \text{Min}\{D_n\} \quad (2)$$

7フレーム、21フレームの特徴を別々の符号帳にし

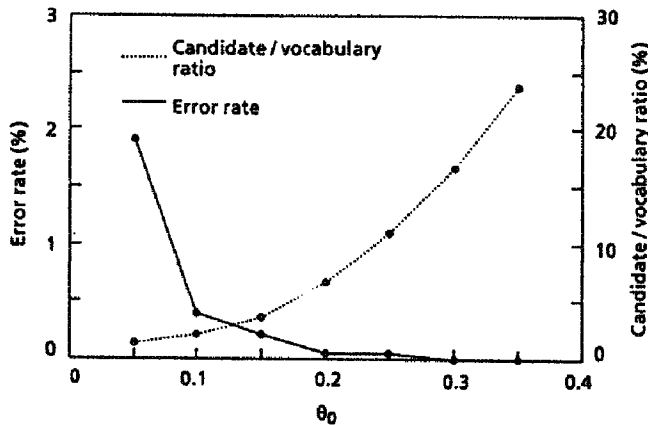


図 4 単語候補の予備選択の結果 (二つの符号帳による距離を組み合わせる場合)

Fig. 4 Preprocessor performance for the multiple codebook method as a function of the threshold bias θ_0 .

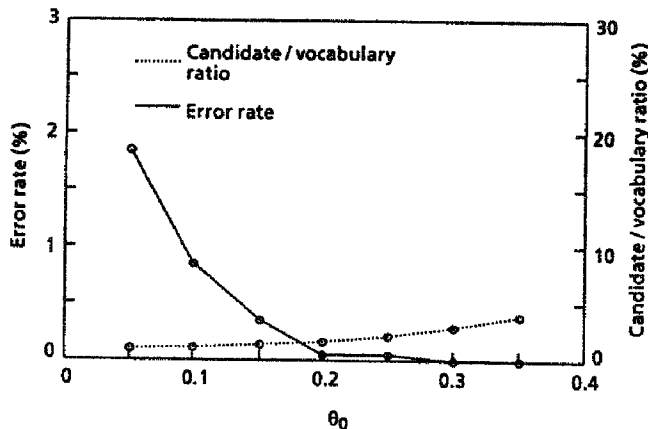


図 5 単語候補の予備選択の結果 (二つの特徴を組み合わせた単一符号帳の場合)

Fig. 5 Preprocessor performance for the single codebook method incorporating hierarchical dynamic features.

て、最適な重みで両者の距離を加算した場合の結果を図 4 に、単一符号帳にした場合の結果を図 5 に示す。いずれの場合も、しきい値バイアス (θ_0) を変えたときに、正しい候補が脱落する誤り率と、選択される候補数の全候補に対する割合を示した。単一符号帳にした方が、候補数を大幅に減らせることがわかる。脱落誤りが 0.05% ($\theta_0=0.2$) のとき、選択される候補数の全候補に対する割合は、複数符号帳のとき 6.61%、単一符号帳のとき 1.67% である。

更に、選択される候補数が、各入力音声に対して 1 単語のみの場合を調べたところ、図 6 に示す結果が得られた。 $\theta_0=0.2$ の条件では、複数符号帳の場合に全入力音声の 28.0%、単一符号帳の場合に全入力音声の 71.8% に対しては 1 単語のみが選択される。これらの

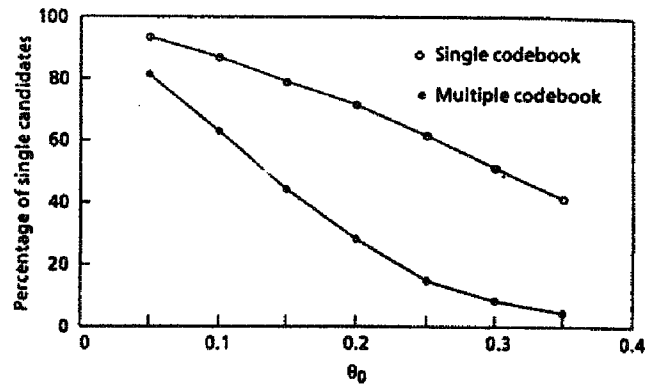


図 6 予備選択によって 1 入力音声に対し単一候補が選択される割合

Fig. 6 Probability of obtaining a single candidate for each input utterance as a function of the threshold bias θ_0 .

場合は、以後の認識処理 (DP 演算) が不要となるので、この数を差し引くと、単一符号帳の場合に認識処理が必要な候補数は、全候補数の 0.95% になる。

3. 音素認識

3.1 方 法

ATR で作成された 5,240 単語データベース (発声者 MAU) に含まれる /b//d//g/ 音声を用いて、動特性を含む符号帳による認識実験を試みた。話頭・話尾切断音節を用いた音声知覚に関する研究⁽¹⁾において明らかにしたように、一般的に人間が音節および音素を知覚するとき最も重要な情報は、子音部から母音部へのスペクトル変化の最も大きい区間にある。この知見に基づき、上記データベースから、/b/, /d/, および /g/ とそれらに後続する母音の境界を中心に一定長の音声区間を切り出し、これを用いて /b//d//g/ の認識実験を行った。

上記データベースには /b/, /d/, /g/ がそれぞれ 445, 382, 512 個含まれるので、それぞれ奇数番目に出現したものを学習サンプルに、偶数番目に出現したものをテストサンプルに用いた。標準化周波数、分析次数などの音声分析条件は単語の場合と同様であるが、音素認識には単語認識よりも時間的にやや細かい情報が必要と考えられるので、フレーム長は 25 ms、フレーム周期は 7.5 ms とした。 Δ ケプストラム、 Δ パワーを求めるフレーム数は、3 フレームと 11 フレームの間で変化させて実験を行った。

フレーム数を変えたときの、 Δ ケプストラムと Δ パワーの /b//d//g/ に関する平均分散を図 7 に示す。 Δ ケプストラムについては、1 次から 10 次までの係数に

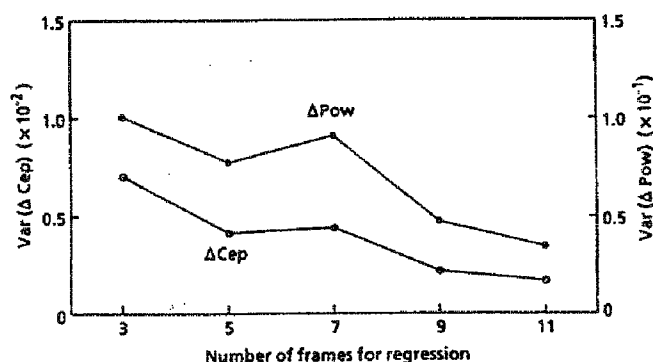


図7 Δ ケプストラムと Δ パワーの/b, d, g/に関する平均分散

Fig. 7 Mean variances of Δ cepstra and Δ power, calculated for various numbers of frames.

表2 音素/b/, /d/, /g/の認識誤り率

		符号帳サイズ			
		32	64	128	256
特徴量	(c)	7.32%	3.89%	3.74%	2.99%
	(c, Δc_s , Δp_s)	—	3.44	1.49	1.35

関して平均した値（距離計算のときの重みの逆数に相当）を示した。

3.2 実験結果

認識および学習に用いる音声区間を、子音・母音境界を中心に26フレーム(195 ms)とし、回帰係数を求めるフレーム数を5フレームとして、各音素を特徴づける符号帳のサイズを32から256の間で変えたときの、種々の符号帳条件における認識誤り率を表2に示す。なお本実験では、各音素に対応する3種類の符号帳を融合した共通符号帳は作成せず、各符号帳を独立に取り扱った。

表2の結果より、符号帳サイズは、ケプストラムのみのとき(c)は64程度、回帰係数を組み合わせるとき(c, Δc_s , Δp_s)は128程度が必要であり、それ以後もサイズを増すに従ってやや誤り率が低下する傾向があることがわかる。ケプストラムのみの場合に比べて、回帰係数を組み合わせることにより、認識誤りの大きな低下が見られ、符号帳サイズを256とすれば1.35%の誤り率が得られる。この値は、ケプストラムのみの場合の1/2以下である。以上の結果に基づき、以下の実験はいずれも、ケプストラムに回帰係数を組み合わせ、符号帳サイズを256として行った。

回帰係数を求めるフレーム数を変えたときの、誤り率の変化を図8に示す。フレーム数としては、5ないし7(37.5~52.5 ms)程度が最適であることがわか

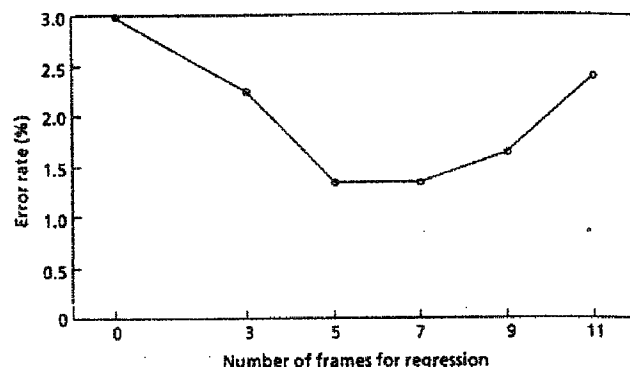


図8 回帰係数を求めるフレーム数を変えたときの/b, d, g/認識誤り率の変化

Fig. 8 /b/, /d/, /g/ recognition error rates as a function of the number of frames for regression analysis.

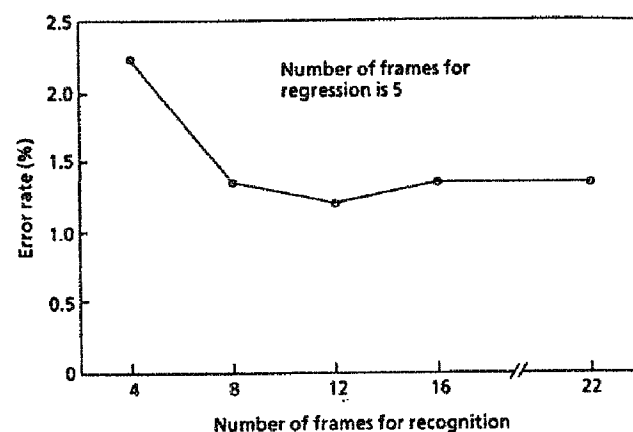


図9 認識に用いるフレーム数を変えたときの/b, d, g/認識誤り率の変化(回帰係数を求めるフレーム数は5フレーム)

Fig. 9 /b/, /d/, /g/ recognition error rates as a function of the speech length.

る。回帰係数を求めるフレーム数を5フレームに固定し、認識に用いるフレーム数(音声区間の両端のそれぞれ2フレームは回帰係数が算出できないので認識には用いないため、最大のフレーム数は、 $26 - 2 \times 2 = 22$ となる)を変えたときの、誤り率の変化を図9に示す。認識に用いるフレーム数は8フレーム(60 ms)あればほぼ十分であり、それ以上の範囲では長さによる誤り率の変化は小さい。細かく見ると、12フレーム(90 ms)のときに誤り率が最小となり、このときの値は1.20%である。

次に、回帰係数を求めるフレーム数が5フレームの場合と、11フレームの場合の特徴量を組み合わせる効果を調べた。単語のときと同様に、別々の符号帳(c, Δc_s , Δp_s)(c, Δc_{11} , Δp_{11})として距離(ひずみ)値で組み合わせる場合と、単一符号帳(c, Δc_s , Δp_s , Δc_{11} , Δp_{11})

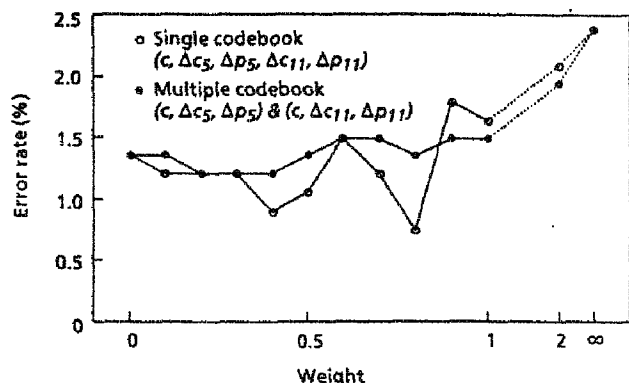


図 10 二つの符号帳による距離を組み合わせる場合と二つの特徴を組み合わせた単一符号帳の場合の /b, d, g/ の認識結果

Fig. 10 Comparison of /b/, /d/, /g/ recognition error rates using the multiple codebook and single codebook methods as a function of the weighting factor combining two distortions.

とする場合の2種類の実験を行った。11フレームの場合の特徴量に対する重みを変えたときの、誤り率の変化を図10に示す。いずれの場合も2種類の特徴量を組み合わせる効果が見られるが、単一符号帳の場合の方がその効果がやや大きい。重みが0.4のときに両符号帳に共通して誤り率はほぼ最小となり、このときの単一符号帳による誤り率は0.90%（認識率99.10%）である。この値は、同じ音声データに関して、HMMやNeural Netを用いた方法によって得られている値^{(9),(10)}とほぼ同等である。

3.3 音素スポッティング

これまでの実験では、人手による音素ラベリングに従って切り出した音声区間を未知音声として用いていたが、より一般的な状況での、スペクトル動特性を表現した符号帳による音声認識法の有効性を確かめるため、/b/, /d/, および/g/のスポッティング実験を行った。

本実験では、まず各音素について、すべての学習サンプルの、子音と後続母音の境界を中心とする12フレームの音声区間を用いて、ケプストラムと単一階層の Δ ケプストラムおよび Δ パワーからなる符号帳(c, Δc_s , Δp_s)を作成した。未知音声の区間を、子音と後続母音の境界を中心とする300msの区間に拡大し、その区間における音素のスポッティングを行った。なお、語頭などで300msの区間の一部が音声区間をはずれる場合は、全区間が音声区間となるように適宜その区間をずらして実験を行った。

その未知音声区間について、18フレーム(135ms)の窓を1フレームずつずらしながら、その窓内の特徴

ベクトルを各音素の符号帳でベクトル量子化した。その窓内で平均化した量子化ひずみが最小となる窓の位置および符号帳を検出し、その最小のひずみを生ずる符号帳の音素が発声されているものと認識した。その認識結果が実際に発声された音素と同じであれば正解とし、検出位置に関しては評価は行わなかった。

認識実験の結果、ラベリングに従って切り出した音声を用いた実験結果とほぼ同程度の、1.05%の平均誤り率が得られた。この結果は、ここで提案した方法が安定に動作することを示している。

4. む す び

音声中の各時点を中心とする長短の異なる長さの区間から抽出した、階層的なスペクトル動特性を用いる新しい音声認識方法を提案し、認識実験により評価を行った。その結果、単語や音素別にこのような特徴ベクトルからなる符号帳を作ることにより、単語や音素の特徴が表現・抽出できることがわかった。入力音声 これらの符号帳でベクトル量子化したときの量子化ひずみを用いることによって、高い精度で認識を行ったり、認識精度を下げずに効率良く候補を予備選択することができる。

単一符号帳と複数符号帳の比較実験の結果、符号帳の大きさが小さい場合は複数符号帳を用いた方がよいが、符号帳を十分大きくできる場合には、単一符号帳を用いた方が、高い認識性能を得ることができることが示された。

このようなスペクトルの過渡特性に着目した認識方法は、発声速度の変化にも強いことが赤木の実験⁽¹¹⁾でも示唆されており、このような符号帳を各音素について用意し、音声区間を並列的にスキャンすることによって、音素スポッティングをベースとした連続音声認識を行うことも可能であると思われる。階層的なスペクトル動特性の表現は、一種のセグメント特徴の表現法と考えることもできる。今後は、動的特徴を含むセグメント符号帳によるVQひずみを入力とする認識方法についても、検討してみたい。

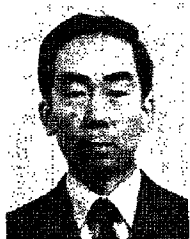
謝辞 音素認識実験に用いた音声データに関して便宜を図って頂いたATR研究所の各位、ならびに実験を手伝って下さった当所松井知子研究員に感謝する。

文 献

- (1) Furui S.: "On the role of spectral transition for speech perception", J. Acoust. Soc. Amer., 80, 4, pp. 1016-1025 (Oct. 1986).

- (2) Furui S.: "Speaker-independent isolated word recognition using dynamic features of speech spectrum", IEEE Trans. Acoust., Speech, Signal Processing, ASSP-34, 1, pp. 52-59 (Feb. 1986).
- (3) Lee K.-F., Hon H.-W and Reddy R.: "An overview of the SPHINX speech recognition system", IEEE Trans. Acoust., Speech, Signal Processing, ASSP-38, 1, pp. 35-45 (Jan. 1990).
- (4) 古井貞熙: "大語彙単語音声認識におけるスペクトル動特性を用いた単語予備選択", 音声研資, SP86-77 (1986-12).
- (5) Furui S.: "A VQ-based preprocessor using cepstral dynamic features for speaker-independent large vocabulary word recognition", IEEE Trans. Acoust., Speech, Signal Processing, ASSP-36, 7, pp. 980-987 (July 1988).
- (6) 寺西立年: "聴覚の時間的側面", 難波精一郎編, 聴覚ハンドブック, 第7章, pp. 276-319, ナカニシヤ出版(1984).
- (7) Furui S.: "On the use of hierarchical spectral dynamics in speech recognition", Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing, Albuquerque, S15a.10 (April 1990).
- (8) Linde Y., Buzo A. and Gray R. M.: "An algorithm for vector quantizer design", IEEE Trans. Commun., COM-28, 1, pp. 84-95 (1980).
- (9) Waibel A., Hanazawa T., Hinton G., Shikano K. and Lang K. J.: "Phoneme recognition using time-delay neural networks", IEEE Trans. Acoust., Speech, Signal Processing, ASSP-37, 3, pp. 328-339 (March 1989).
- (10) 平田好充, 橋本泰秀, 中川聖一: "混合連続出力分布 HMM を用いた有声破裂音の識別", 1989 信学秋季全大, A-18.
- (11) Akagi M. and Tohkura Y.: "On the application of spectrum target prediction model to speech recognition", Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing, New York, S3.11 (April 1988).

(平成2年8月29日受付, 12月3日再受付)



古井 貞熙

昭43東大・工・計数卒。昭45同大学院修士課程了。同年NTT電気通信研究所入社。以後、同研究所において、音声認識、話者認識、音声知覚などの基礎研究に従事。昭53～54ベル研究所客員研究員。現在

NTTヒューマンインタフェース研究所音声情報研究部長。工博、昭50米沢賞、昭63論文賞、平2著述賞、昭60、62音響学会論文賞、平1科学技術庁長官賞、IEEE ASSP Society Senior Award各受賞。著書「デジタル音声処理」「Digital Speech Processing, Synthesis, and Recognition」「音声情報工学」(分担)、「音の科学」(分担)など。IEEE、日本音響学会各会員。

Continuous Speech Recognition Using a Dependency Grammar and Phoneme-Based HMMs

Sho-ichi MATSUNAGA†, Shigeru HOMMA††, Shigeki SAGAYAMA†††
and Sadaoki FURUI†, *Members*

SUMMARY This paper describes two Japanese continuous speech recognition systems (system-1 and system-2) based on phoneme-based HMMs and a two-level grammar approach. Two grammars are an intra-phrase transition network grammar for phrase recognition, and an inter-phrase dependency grammar for sentence recognition. A joint score, combining acoustic likelihood and linguistic certainty factors derived from phoneme-based HMMs and dependency rules, is maximized to obtain the best sentence recognition results. System-1 is tuned for sentences uttered phrase-by-phrase and system-2 is tuned for sentence utterances, to make the amount of computation practical. In system-1, two efficient parsing algorithms are used for each grammar. They are a bi-directional network parser and a breadth-first dependency parser. With the phrase-network parser, input phrase utterances are parsed bi-directionally both left-to-right and right-to-left, and optimal Viterbi paths are found along which the accumulated phonetic likelihood is maximized. The dependency parser utilizes efficient breadth-first search and beam search algorithms. For system-2, we have extended the dependency analysis algorithm for sentence utterances, using a technique for detecting most-likely multi-phrase candidates based on the Viterbi phrase alignment. Where the perplexity of the phrase syntax is 40, system-1 and system-2 increase phrase recognition performance in the sentence by approximately 6% and 14%, showing the effectiveness of semantic dependency analysis.

1. Introduction

In Japanese sentences, the phrase order is much less constrained than in English. On the other hand, the word order within phrases (called "BUNSETU") is very regular, and the sentence structure is ordered by semantic dependency (called "KAKARIUKE") between phrases. Syntactic constraints are useful in recognizing specific tasks or short-duration utterances. However, particularly in sentence recognition for phrase-order-free languages such as Japanese, semantic constraints are more powerful than sentence syntactic constraints. Thus, it is considered that combining different types of these grammars is practical for linguistic processing of Japanese speech recognition. Recently, several dependency analysis algorithm have

been developed⁽¹⁾⁻⁽³⁾.

On the other hand, the hidden Markov model (HMM) was introduced, and achieved good results in speech recognition^{(4),(5)}. Our goal is to develop an effective method of combining high-level linguistic knowledge and refined phonetic models to improve speech recognition results. Therefore, we have developed two Japanese continuous speech recognition systems that obtain the most likely sentence results taking account of acoustic, syntactic, and semantic factors based on phoneme-based HMMs and a two-level grammar approach⁽⁶⁾. Each system uses two grammars: an intra-phrase transition network grammar for phrase recognition, and an inter-phrase dependency grammar for sentence recognition. The former is a syntactic grammar, and the latter is a semantic and loose syntactic grammar. The dependency grammar is compatible with the case grammar. Thus, semantic certainty factor is determined taking into account grammatical cases.

System-1 is tuned for sentences uttered phrase-by-phrase and system-2 for sentence utterances, to make the amount of computation practical. In system-1, two efficient parsing algorithms are used for each grammar. They are a bi-directional network parser and a breadth-first dependency parser. With the phrase-network parser, input phrase utterances are parsed bi-directionally both left-to-right and right-to-left, and optimal Viterbi paths are found along which the accumulated phonetic likelihood is maximized. The parser generates a phrase matrix. With the dependency parser, inter-phrase dependency structures within a sentence are analyzed. The joint score, obtained by combining accumulated phonetic likelihood and semantic certainty factor, is maximized to obtain the solution on this matrix. The dependency parser utilizes efficient breadth-first search and beam search algorithms.

For system-2, we have extended the dependency analysis algorithm for sentence utterances, using a technique for detecting most-likely multi-phrase candidates based on the Viterbi phrase alignment. Taking account of pauses between phrases, syntactic constraints of sentences are represented by multi-connection of these networks. The network parser

Manuscript received January 10, 1991.

Manuscript revised March 25, 1991.

† The authors are with NTT Human Interface Laboratories, Musashino-shi, 180 Japan.

†† The author is with NTT Tohoku Software Plan and Produce Center, Sendai-shi, 980 Japan.

††† The author is with ATR Interpreting Telephony Research Laboratories, Kyoto-fu, 619-02 Japan.

frame-synchronously parses input sentence utterances to get a lattice of the multi-phrase candidates and their likelihoods for the selected phrase boundaries extracted by the Viterbi algorithm, and generates a phrase lattice. The dependency parser also analyzes inter-phrase dependency structures on the lattice.

The approach of these two systems is very suitable for speech understanding systems since it can use semantic dependency structures.

2. Two-Level Grammar Approach

2.1 Intra-Phrase Grammar

A Japanese phrase is composed of a stem part and a suffix part. The stem part is a verb, a noun, or an adjective. The suffix part is composed of suffixes such as auxiliary verbs and case markers called "JOSHI". The connection of these parts is so regular that phrase syntax is represented by networks. However, there are many kinds of connection rules. Therefore, to cover concisely this variety, the syntactic structure is represented by transition networks (TNs) composed of many sub-networks.

2.2 Dependency Structure Grammar

After the phrase likelihood has been calculated, recognition results are represented in a matrix form or in a lattice form. When analyzing dependency relationships, the parser extracts the most likely sentence by taking into account the phonetic likelihood of phrase candidates and the linguistic certainty factor of semantic inter-phrase dependency relationships. The parser also obtains the dependency structure that corresponds to the semantic structure of the extracted sentence.

Dependency grammar is based on semantic dependency relationships between phrases. The syntactic rules satisfy the following two constraints. First, every phrase, except the last phrase, must modify one and only one later phrase. This modification is called a dependency relationship or dependency structure. It is represented by an arc frequently. Second, modification arcs between phrases must not cross.

Because dependency structure grammar is compat-

ible with case grammar, the semantic certainty factor of the dependency structure is provided using case structure. This is described in detail in reference⁽⁹⁾. Certainty factor is composed of two categories. One is the certainty factor associated with dependency relationships of only the modifier and modificant phrases: disagreement between the semantic primitive of the modifier and that required by the modificant, disagreement between the case of modifier and that required by the modificant, incorrect idiomatic expression and so on. The other is the certainty factor associated with all the dependency structure of the phrase sequence: lack of a phrase of the obligatory case required by the modificant, modification of the same phrase by different phrases having the same case category, complexity of dependency structure and so on. The certainty factor values for these items are given as penalties (negative values) heuristically.

3. Speech Recognition System-1

A block diagram of system-1 is shown in Fig. 1. Input sentences are uttered phrase by phrase. After feature parameter extraction from the utterance, the parameter sequence is converted into a vector code sequence. Next, phonetic likelihood is calculated for every possible duration to obtain likelihood matrices for phoneme candidates, based on HMMs. Phonetic duration time can be easily controlled by giving maximum and minimum duration times to each phoneme using the matrix. Next, phrase likelihood is calculated based on the phonetic likelihood matrices. The verb and adjective entries in the dictionary have grammatical cases, and the noun entries are accompanied by semantic primitives. Next, the top candidates of each likelihood are generated in a matrix form. The number of candidates is optional. Finally, using this matrix and the dependency grammar, the parser extracts the most likely sentence of a phrase sequence and its dependency structure.

3.1 Speech Recognition of Phrases

If the phrase parser expands the transition networks to a single network in phrase recognition proc-

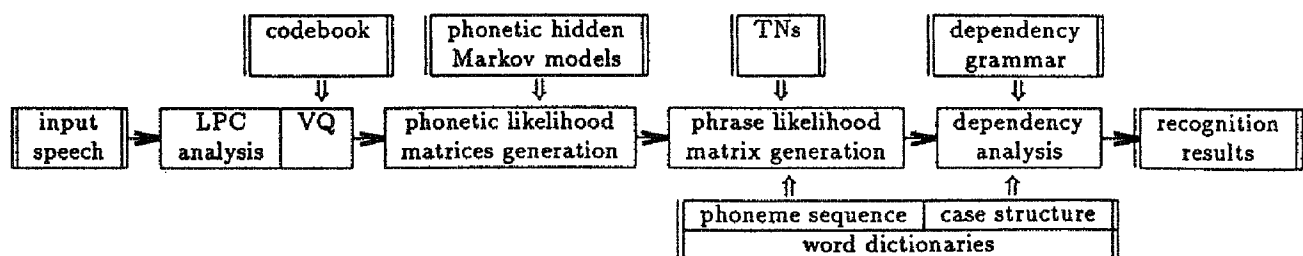


Fig. 1 Block diagram of the recognition system (system-1) for continuous speech uttered phrase by phrase.

ess, the network becomes too large to handle and too time consuming because of the complexity of phrase structure. To overcome these problems, a bi-directional parser was developed. Sagayama et al. developed the original bi-directional approach for connected word recognition using the DP algorithm⁽⁷⁾. Our parser parses the stem part left-to-right, and suffix part right-to-left. Bottom-up parsing is carried out for the stem and the last suffix of the suffix part, and top-down parsing is carried out for inner suffix called "KAT-SUYOO". By calculating the top-most m candidates for each sub-network, the top-most m phrase candidates are generated based on the Viterbi algorithm.

A simple example of phrase syntax and flow of parsing are shown in Fig. 2. S and B correspond to syntax networks, and A , C , D and E correspond to word dictionaries. Processing flow is $A \rightarrow D \rightarrow E \rightarrow C \rightarrow B \rightarrow S$, and its processing sequence is shown in the figure. First, the stem words, corresponding to A , are parsed left-to-right from the beginning of speech. Next, the last suffixes, corresponding to D and E , are processed right-to-left from the end of speech. The parsing result of B is generated by combining the results of D , E and C . The likelihood calculation of C , namely likelihoods of $c1$ and $c2$, is carried out at every top-down scan of C . Finally, the phrase recogni-

tion result is generated by combining the results of A and B . In the figure, \rightarrow , \leftarrow , \uparrow , \downarrow indicate the left-to-right process, the right-to-left process, the bottom-up control, the top-down control, respectively.

3.2 Performance of a Bi-directional Parser

The performance of the bi-directional parser (BD method) is evaluated by comparing two-level dynamic programming approach using the Cocke-Kasami-Younger algorithm (TD method)^{(8),(9)} and frame synchronous parsing approach (FS method)^{(10),(11)}. The two levels of the TD method are calculation stages of phonetic likelihood and that of phrase likelihood. Each parser's performance is slightly different when each HMM is learnt using the Baum-Welch algorithm. The performance of TD using a forward trellis computation is the highest. That of BD using the forward computation for intra-phoneme likelihood derivation and the Viterbi algorithm for inter-phoneme is next. That of FD is third using the Viterbi algorithm only. Processing amount of the forward trellis is much larger than that of the Viterbi algorithm.

Comparison experiments are carried out using input phrases whose duration is about one second. Three kinds of linguistic constraints are used whose number of word entries are 360, 130 and 20. The top-five candidates are generated in each parsing. Processing time ratio is shown in Table 1. These results show that the BD method is more effective than the FS method when vocabulary size is larger. This is because the BD method looks up the phoneme likelihood matrices which are calculated in advance. Furthermore, the TD method needs much more computation than the other methods, and it is considered that the BD method is one of the most useful parsing methods using phoneme-based HMMs for Japanese phrase recognition owing to recognition performance and parsing amount.

3.3 Parser for Dependency Structure Grammar

Dependency parsing is equivalent to solving the following objective function using the constraints of dependency structure grammar.

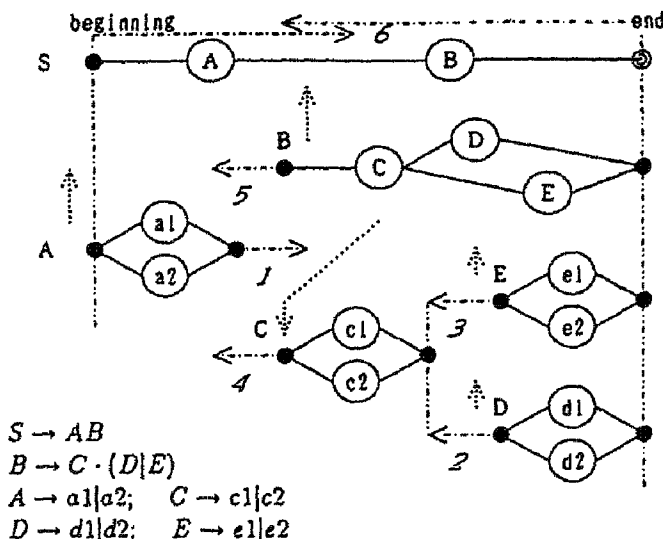


Fig. 2 Example of phrase recognition process.

Table 1 Processing amount ratio for phrase recognition.

parsing approach	BD	FS	TD
HMM recognition method (intra-phoneme)	trellis	Viterbi	trellis
(inter-phoneme)	Viterbi	Viterbi	Viterbi (Trellis is possible)
performance order	2	3	1
word entries (360 words)	16	18	626
word entries (130 words)	12	12	-
word entries (20 words)	5	1	-

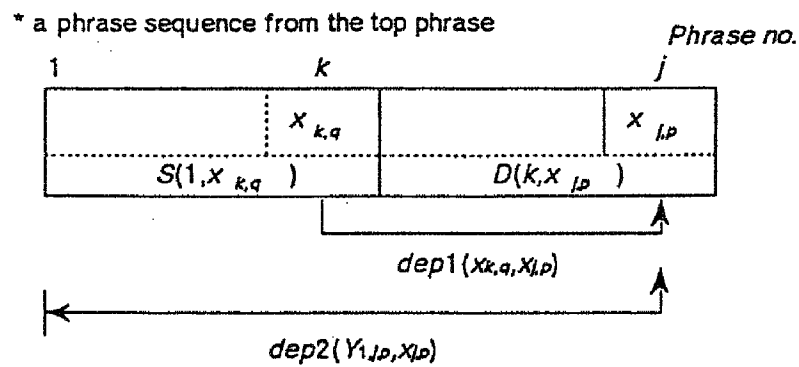
Processing amount is normalized by that of FS method using 20 words-dictionary.

$$T = \max_{(p)} \left[\sum_{j=1}^N c(x_{j,p}) + \max_Y \sum_{j=1}^N \text{dep}(w_{1,j-1}, x_{j,p} | Y_{1,j,p}) \right] \quad (1)$$

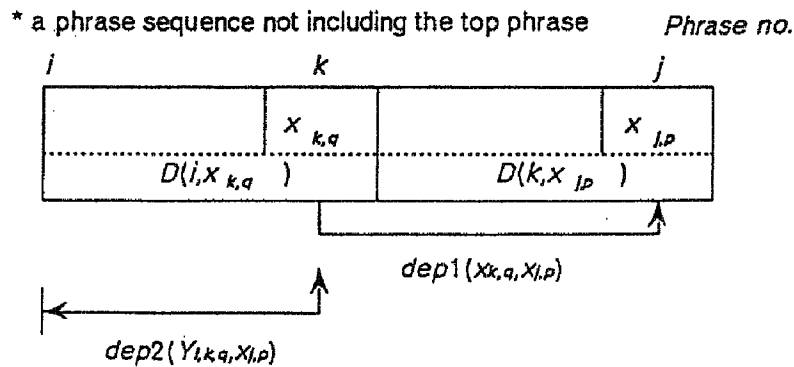
where $1 \leq j \leq N$, $1 \leq p \leq M$, N is the number of input phrases, M is the maximum number of recognition candidates for each uttered phrase, $x_{j,p}$ is a candidate of the j -th input phrase with the p -th best likelihood, and $c(x_{j,p})$ is its log-likelihood. A phrase sequence with one phrase candidate for each i -th to j -th input phrase and whose last phrase is $x_{j,p}$ is denoted by $X_{i,j,p}$. $Y_{i,j,p}$ is one of the dependency structures of $X_{i,j,p}$, and $w_{i,j-1}$ is the set of phrases that modify $x_{j,p}$ in

the sequence $X_{i,j,p}$. Here, $\text{dep}(w, x | Y)$ is the linguistic certainty factor of dependency relationships between w and x taking Y into account. The first item of the term on the right in Eq.(1) is the summation of phonetic likelihoods of the hypothesized sentence composed of its phrase sequence, and the second item is the summation of linguistic certainty factor. Maximizing Eq.(1) gives the sentence and its dependency structure as the speech recognition result.

To solve Eq.(1) effectively, a fast parsing algorithm using breadth-first search and beam search was used⁽⁹⁾. This algorithm is based on the fundamental algorithm⁽¹⁾. Although it offers sub-optimal solutions, it is practical because it requires much less processing



$$S(1, x_{j,p}, r) = r^{\text{th}} \max_{k,q,r1,r2} [S(1, x_{k,q}, r1) + D(k, x_{j,p}, r2) + \text{dep1}(x_{k,q}, x_{j,p}) + \text{dep2}(Y_{1,j,p}, x_{j,p})]$$



$$D(i, x_{j,p}, r) = r^{\text{th}} \max_{k,q,r1,r2} [D(i, x_{k,q}, r1) + D(k, x_{j,p}, r2) + \text{dep1}(x_{k,q}, x_{j,p}) + \text{dep2}(Y_{i,k,q}, x_{j,p})] \quad (i \neq 1)$$

Fig. 3 Illustration of deriving the recurrence relation among the objective function $i \leq k \leq j-1$, $1 \leq q \leq M$, and $1 \leq r, r1, r2 \leq L$. $r, r1$ and $r2$ indicate the beam ranks, L is the maximum number of beams, $S(1, x_{j,p}, r)$ and $D(i, x_{j,p}, r)$ are the r -th value of the element whose phrase sequence is $X_{i,j,p}$, and the dependency structure is $Y_{i,j,p}$. Here, $r^{\text{th}} \max[\cdot]$ is a function for deriving the r -th best value.

than the depth-first search⁽¹²⁾ that derives the optimal solutions.

The breadth-first algorithm is formulated as follows. First, $dep(w, x|Y)$ can be divided into two terms:

$$dep(w_{1,j-1}, x_{j,p}|Y_{1,j,p}) \\ = \sum_{x \in w_{1,j-1}} dep1(x, x_{j,p}) + dep2(Y_{1,j,p}, x_{j,p}) \quad (2)$$

where $dep1$ is the certainty factor associated with dependency relationships of only the modifier and modificant phrases, and $dep2$ is the certainty factor associated with $Y_{1,j,p}$.

This algorithm uses beam-search to recursively derive $S(1, x_{j,p})$, the objective function's value of a phrase sequence from the top phrase to phrase $x_{j,p}$, as well as $D(i, x_{j,p})$, the value of a phrase sequence not including the top phrase ($i \neq 1$) as shown in Fig. 3. In these equations, each dependency structure for evaluation of $dep2$ is different. When the equations are calculated, $Y_{i,j,p}$ is stored for later use evaluating $dep2$. Initial values are given using phonetic likelihood based on HMMs.

After calculating the recurrence relation, the value

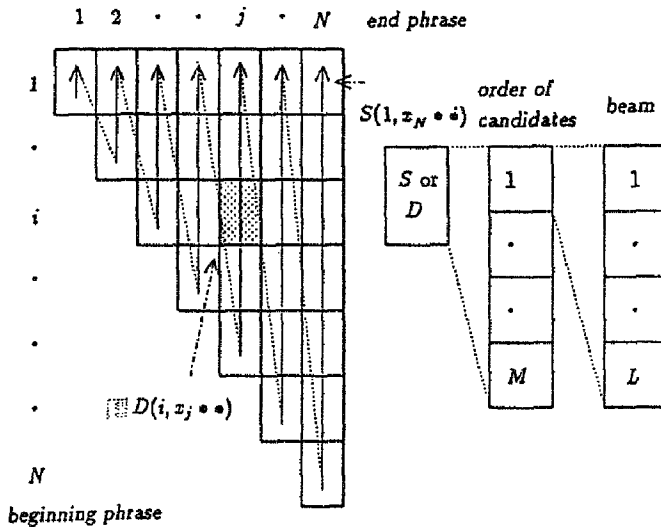


Fig. 4 Configuration of parsing table.

of the objective function is obtained:

$$T = \max_p [S(1, x_{N,p}, 1)] \quad (3)$$

where $1 \leq p \leq M$. The best sentence and its dependency structure are given through $Y_{1,N,p}$ where p maximizes Eq. (3). The parsing table is shown in Fig. 4. In Fig. 4, the first row corresponds to S , and others correspond to D . The phrase sequence for the first to N -th phrase corresponds to the right-most top cell. Each cell is composed of M sub-cells for the number of candidates, and each sub-cell is composed of L sub-cells for the beam width. The arrows show the sequence of calculating the recurrence relation. The processing amount order for this algorithm is $O(N^3 M^2 L^2)$.

4. Speech Recognition System-2

Figure 5 shows a block diagram of system-2. Inputs are sentence utterances. Phrase likelihood is calculated frame-synchronously using phoneme-based HMMs and transition networks. This calculation is based on the Viterbi algorithm, and yields several best phrase sequences for each frame whose last phrase is different from that of other sequences. The last phrase of each sequence for each end-frame is stored in lattice form. This lattice is reduced based on Viterbi phrase alignment. Then, the parser extracts the most likely sentence on the lattice.

4.1 Speech Recognition of Phrase Sequences

The syntactic constraints of sentences are governed

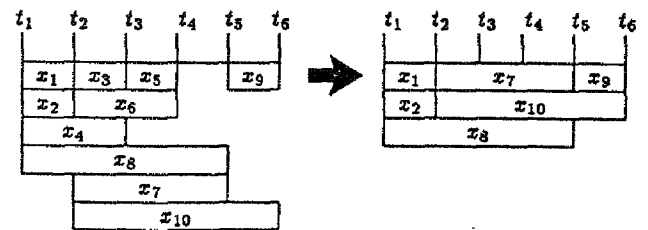


Fig. 6 Example of a reduced phrase lattice. x shows a phrase, and t shows a frame or time.

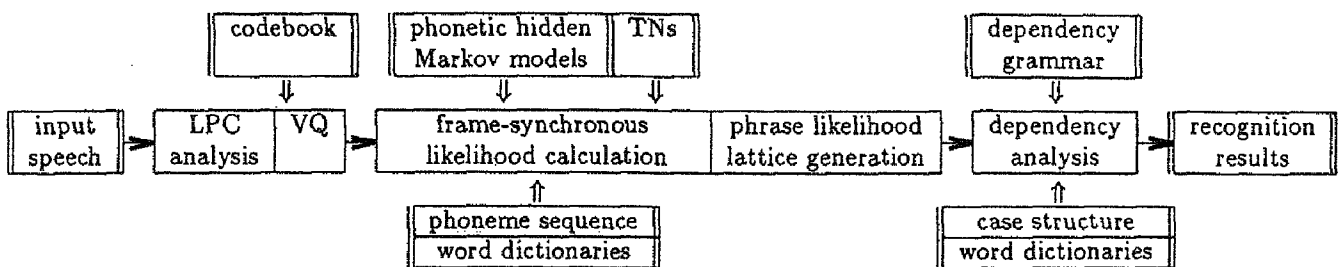


Fig. 5 Block diagram of the recognition system (system-2) for sentence speech.

by connection of phrase networks taking account of pauses between phrases. When the frame-synchronous parser parses the input from left to right using phoneme-based HMMs and networks based on the Viterbi algorithm, this calculation generates M -best phrase sequences whose last phrases are different from each other. The last phrases are stored in lattice form. In other words, there are M -phrases for each end-frame and the total number of phrase candidates in the lattice is approximately NM , if input is N -frame speech. After the whole lattice has been generated, significant phrase boundaries, which make consecutive phrase sequences are back-traced from the end of the speech to the beginning according to Viterbi phrase alignment. Phrase boundary candidates are then selected and a reduced lattice is generated. A simple example, where M is 2 and N is 6, is shown in Fig. 6. The left side of this figure is the original lattice. Three sets, that is (t_1, t_2, t_6) , (t_1, t_5, t_6) and (t_1, t_2, t_5, t_6) , can generate consecutive phrase sequences, and the new reduced lattice is on the right.

4.2 Dependency Parser for Sentence Speech

Consecutive phrases, whose number is H , of a sentence on the phrase lattice are given as follows.

$$x_{j_1, l_1, p_1} \cdots x_{j_H, l_H, p_H}$$

where $1 \leq h \leq H$, $1 \leq j \leq N$, $1 \leq p \leq M$, N is the number of frames of the input sentence, and M is the number

of phrase candidates of each segment. Here, j_h, l_h, p_h is abbreviated to j, l, p . The term $x_{j, l, p}$ is a candidate of the $j-l$ -th to j -th frame with the p -th best likelihood. The other terms shown in Sect. 3.3 are also transformed by adding the subscript concerned with phrase length l .

Lattice parsing selects the best phrase sequence in the lattice taking account of HMM likelihood and linguistic certainty factor. It is equivalent to solving the following objective function:

$$T = \max_{H, (p)} \left[\sum_{h=1}^H c(x_{j, l, p}) + \max_Y \sum_{h=1}^H dep(w_{1, j}, x_{j, l, p} | Y_{1, j, l, p}) \right] \quad (4)$$

The breadth-first algorithm, which was described in Sect. 3.3, has been extended to sentence speech like other algorithms^{(13), (14)}. Our original algorithm shown in Table 2 uses a recurrence relation, and deals with the linguistic certainty factor associated with dependency structure. This is different from the other algorithms.

After calculating the recurrence relation, the value of the objective function is obtained:

$$T = \max_{l, p} [S(1, x_{N, l, p}, 1)] \quad (5)$$

where $1 \leq p \leq M$ and $1 \leq l \leq N$. The best sentence and its dependency structure are given through $Y_{1, N, l, p}$ where p and l maximize Eq. (5). Although the processing amount order for this algorithm is $O(N^5 M^2 L^2)$, it is not practical.

Table 2 Original parsing algorithm for sentence speech.

{1} : Loop for the end frame of the end phrase of the partial sequence DO {2} to {6} for $j = 2, 3, \dots, N$
{2} : Loop for the beginning frame of beginning phrase DO {3} to {6} for $i = j - 1, \dots, 1$
{3} : Loop for the length of end phrase DO {4} to {6} for $l = 1, \dots, i$
{4} : if $j - i = l$, set the initial value
{5} : Loop for the end phrases DO {6} for $p = 1, 2, \dots, M$
{6} : Computation of recurrence relation (Loop for the end phrase of the former sequence) {6-1} : DO {6-2} to {6-5} for $k = i + 1, \dots, j - l$ {6-2} : DO {6-3} to {6-5} for $g = 1, \dots, k - i$ {6-3} : DO {6-4} to {6-5} for $q = 1, 2, \dots, M$ (Loop for the beam width) {6-4} : DO {6-5} for $r1 = 1, 2, \dots, L$ {6-5} : for $r2 = 1, 2, \dots, L$ * Evaluation of $S(1, x_{j, l, p}, r)$ or $D(i, x_{j, l, p}, r)$ taking account of $Y_{1, j, l, p}$ or $Y_{i, k, g, q}$ * Store of $Y_{i, j, l, p}$
{7} : Acquisition of the parsing results * Detection of values, p and l , maximizing Eq.(5) * Acquisition of the phrase sequence and its dependency structure using $Y_{1, N, l, p}$

1 or i	$k-g$	k	$j-l$	frame j
	$x_{k, g, q}$		$x_{j, l, p}$	
$S(1, x_{k, g, q})$ or $D(i, x_{k, g, q})$		$D(k, x_{j, l, p})$		

4.3 Further Reduction of Processing Based on Viterbi Phrase Alignment

After selection of phrase boundaries, the reduced phrase lattice is generated. In this case, the parsing algorithm is extended as follows. Loops {3} for phrase length N and {5} for the number of fixed-length phrase candidates M in Table 2 are reduced to one loop for the number of free-length candidates \tilde{M} . In the same way, loops {6-2} and {6-3} are reduced to a loop. Then, the processing amount order for this extended algorithm is $O(\tilde{N}^3 \tilde{M}^2 L^2)$. If $\tilde{N} = N/10$, $\tilde{M} = M$, and the length of the input sentence is about 310 frames, then the parser reduces computation to approximately 10^{-8} of the amount for the original parser proposed in Sect. 4.2. The parsing table is similar to that of Fig. 4. Each cell in the parsing table is composed of \tilde{M} sub-cells for the number of candidates, and each sub-cell is composed of L sub-cells for the beam-width.

5. Speech Recognition Experiments

Input utterances were sampled at a rate of 12 kHz. One frame was extracted every 10 ms with a 30 ms

Hamming window and converted into 34 acoustic feature parameters: power, 16 LPC cepstra, Δ power, and 16 Δ LPC cepstra.

In the training process, 216 phonetically balanced words were used. These utterances were manually labeled using 25 phoneme symbols including silence. Each phoneme was modeled by HMM and had 4 states and 7 transition paths. The parameter sequence was converted into a vector code sequence and a vector codebook composed of 256 prototype vectors was generated. Each HMM was trained using the forward-backward algorithm, and the code sequence for training was cut out based on the phonetic labels. Output probabilities were floored after training.

In the testing process, acoustic feature parameters of the input were generated in the same manner. These parameters were converted into a code sequence using the same speaker's codebook generated in the training process.

Talker-dependent preliminary word recognition tests were carried out with 216 words uttered by ten speakers. The system attained a word recognition rate of 98.4%.

Recognition tests for system-1 were then performed on 100 sentences (including 668 phrases in an essay) uttered phrase-by-phrase by the same speakers. The word dictionary had 360 entries and the perplexity of the phrase syntax was 40. Certainty factors of dependency relationships were empirically determined through the analysis of technical literature. If the training set is 216 words, the system attained a phrase recognition rate of 80.8% using only the intra-phrase syntactic parser. The dependency parser increased this rate to 86.8% as shown in Table 3.

Finally, to evaluate system-2, sentence speech recognition was tested for 71 sentences (including 418 phrases) uttered by two speakers, one male and one female. In this experiment, 2 training sets were used: 216 phonetically balanced words, and combination of 216

words and 29 sentences. The maximum number of each phrase segment \bar{M} was 5, and the beam-width L was 8. Word dictionary, word perplexity, and certainty factors were the same as in the preceding tests. Table 4 shows that the dependency parser increased the phrase recognition rate of 69.2% to 83.1% and increased the sentence recognition rate of 11% to 40%, if the training set was the combination of 216 words and 29 sentences. These results show the effectiveness of semantic dependency analysis.

6. Conclusion

This paper described two Japanese continuous speech recognition systems using an intra-phrase transition network grammar and an inter-phrase dependency grammar. System-1 was tuned for sentences uttered phrase-by-phrase and system-2 for sentence utterances, to make the amount of computation practical. In system-1, the best sentence was efficiently determined from input utterances using a bi-directional network parser and a breadth-first dependency parser. In system-2, it was determined using a frame-synchronous parser and an extended dependency parser combined with the Viterbi phrase alignment technique. Where the perplexity of the phrase syntax is 40, system-1 and system-2 attained phrase recognition performance in the sentence by approximately 86.8% and 83.1%, showing the effectiveness of the two systems, especially the inter-phrase dependency grammar.

Further development is needed to refine the linguistic certainty factor based on a statistical method and phoneme-based models based on continuous HMMs that take context dependency into account.

Acknowledgement

The authors wish to express their appreciation to Prof. Masaki Kohda of the University of Yamagata and to Kiyohiro Shikano of NTT Laboratories for their invaluable discussions.

References

- (1) Ozeki K.: "A multi-stage decision algorithm to select optimum bunsetsu sequences based on degree of kakariuke-dependency", Trans. IEICE Japan, **J70-D**, 3, pp. 601-609 (March 1987).
- (2) Nakagawa S. and Ito T.: "Recognition of spoken Japanese sentences using mono-syllable units and backward kakariuke parsing algorithm", Trans. IEICE Japan, **J70-D**, 12, pp. 2469-2478 (Dec. 1987).
- (3) Matsunaga S.: "A breadth-first technique of dependency relationships analysis for minimal phrase speech recognition", Trans. IEICE Japan, **J71-D**, 10, pp. 1966-1976 (Oct. 1988).
- (4) Bahl L. R., et al.: "Further results on the recognition of a continuously read natural corpus.", Proc. ICASSP (1980).

Table 3 Speech recognition results (system-1, phrase-by-phrase speech).

data	216 words		668 phrases	
	training		test	
dependency analysis	-		without	with
recognition rate (top 1 [%])	99.9	98.4	80.8	86.8
(top 5 [%])	100	100	96.8	-

Table 4 Speech recognition results (system-2, sentence speech).

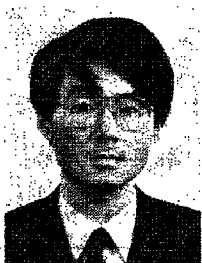
test data	phrase recognition rate [%]			
	71 sentences (418 phrases)			
HMM training data	216 words		216 words + 29 sent.	
dep. analysis	without	with	without	with
recog. rate(male)	45.5 (0)	55.3 (3)	63.6 (4)	78.9 (34)
recog. rate(female)	54.5 (11)	75.4 (32)	74.8 (18)	87.3 (46)
average	50.0 (6)	65.3 (18)	69.2 (11)	83.1 (40)

Each () indicates sentence recognition rate. [%]

- (5) Lee K-F. and Hon H-W.: "Large vocabulary speaker-independent continuous speech recognition using HMM", Proc. ICASSP, pp. 123-126, New York (1988).
- (6) Matsunaga S., et al.: "A continuous speech recognition system based on a two-level grammar approach", Proc. ICASSP, pp. 589-592, Albuquerque (1990).
- (7) Sagayama S. and Nakatsu R.: "Reversed DP matching applied to speech recognition of connected words", Proc. ASJ autumn meeting, 2-2-18, pp. 547-548 (Oct. 1979).
- (8) Sakoe H.: "Two-level DP-matching—A dynamic programming based pattern matching algorithm for connected word recognition", IEEE Trans. Acoust., Speech & Signal Process., ASSP-27, 6, pp. 588-595 (Dec. 1979).
- (9) Levinson S. E.: "Structural methods in automatic speech recognition.", Proc. IEEE, pp. 1625-1650 (Nov. 1985).
- (10) Bridle J., et al.: "An algorithm for connected word recognition.", Proc. ICASSP, pp. 899-902 (May 1982).
- (11) Sakoe H., et al.: "A high speed DP-matching algorithm based on frame synchronization, beam search and vector quantization.", Trans., IEICE Japan, J71-D, 9, pp. 1650-1659 (Sept. 1988).
- (12) Matsunaga S. and Kohda M.: "Speech recognition of minimal phrase sequence taking account of dependency relationships between minimal phrases", Trans. IEICE Japan, J70-D, 11, pp. 2102-2107 (Nov. 1987).
- (13) Ozeki K.: "A multi-stage decision algorithm to select optimum kakariuke structures from bunsetsu lattice", Trans. IEICE Japan, J70-D, 12, pp. 2621-2629 (Dec. 1987).
- (14) Kohda M.: "An algorithm for optimum selection of phrase sequence from phrase lattice" IECE Technical Report, NLC86-14 (Dec. 1986).



Sho-ichi Matsunaga received the B. S. and M. S. degrees in information science engineering from Kyushu University, Fukuoka, Japan in 1979 and 1981. Since 1981, he has been associated with Nippon Telegraph and Telephone Corporation, working on automatic speech recognition. Mr. Matsunaga is a member of the Acoustical Society of Japan.



Shigeru Homma was born in Tochigi Prefecture, Japan on November 18, 1958. He received the B. E. degree in electrical engineering, and the M. E. degree in information engineering from Tohoku University, Sendai, Japan in 1981 and 1983. From 1983 to 1987, he worked at the Yokosuka Electrical Communication Laboratories of Nippon Telegraph and Telephone Corporation in Kanagawa, Japan, where he did research on natural language processing of Japanese. From 1987 to 1990, he worked at the NTT Human Interface Laboratories in Tokyo, Japan, where he was engaged in speech recognition research. Since 1990, he has been with the NTT Tohoku Software Plan and Produce Center in Sendai, Japan, where he is currently Head of the Development Section. Mr. Homma is a member of the Information Processing Society of Japan and the Acoustical Society of Japan.



Shigeki Sagayama was born on 12 May 1948 in Hyogo Prefecture, Japan. He received the B. S. and M. S. degrees in mathematical engineering and instrumentation physics from Tokyo University in Tokyo, Japan in 1972 and 1974. After joining the Nippon Telegraph and Telephone Public Corporation in 1974, he conducted research in speech analysis, speech recognition, and speech synthesis at the NTT Telecommunications Laboratories and, later, at NTT Human Interface Research Laboratories. Since 1990, he has been with the ATR Interpreting Telephony Research Laboratories in Seika-cho, Kyoto, doing speech recognition and speech synthesis research. Mr. Sagayama is a member of the Institute of Electrical and Electronics Engineers and the Acoustical Society of Japan.



Sadaoki Furui received the B. S., M. S., and Ph. D. degrees in mathematical engineering and instrumentation physics from Tokyo University, Tokyo, Japan in 1968, 1970, and 1978, respectively. After joining the Electrical Communications Laboratories, Nippon Telegraph and Telephone Corporation in 1970, he studied the analysis of speaker characterizing information in the speech wave, its application to speaker recognition as well as inter-speaker normalization and adaptation in speech recognition, the vector-quantization-based speech recognition algorithm, the spectral dynamic features for speech recognition, and the analysis of speech perception mechanism. He is currently the Executive Manager of Speech and Acoustics Laboratory at NTT Human Interface Laboratories. From December 1978 to December 1979 he was with the Staff of the Acoustics Research Department at Bell Laboratories, Murray Hill, New Jersey, as a visiting researcher working on speaker verification. Dr. Furui is a member of the Institute of Electrical and Electronics Engineers and the Acoustical Society of Japan. He received the Yonezawa Prize and the Paper Award from the IEICE in 1975 and 1988, respectively, and the Sato Paper Awards from the ASJ in 1985 and 1987, respectively. He received the Senior Award from the IEEE ASSP Society and the Achievement Award from the Minister of Science and Technology, both in 1989. He also received the Book Award from the IEICE in 1990. He is the author of "Digital Speech Processing, Synthesis, and Recognition" (Marcel Dekker, 1989) in English and "Digital Speech Processing" (Tokai University Press, 1985) in Japanese.

A TOP-DOWN DISCOURSE ANALYSIS IN A SPEECH DIALOGUE SYSTEM

Yasuhisa NIIMI and Yutaka KOBAYASHI

Department of Electronics and Information Science, Kyoto Institute of Technology
 Matsugasaki, Sakyo-ku, Kyoto 606, Japan

This paper describes a method for the discourse analysis performed in the speech dialogue system we are developing. The purpose of the analysis is to provide the system with top-down discourse hypotheses. These are translated into linguistic constraints and used to reduce a search space in recognizing an utterance to come next. This effect of the discourse analysis has been proved by a simulation of the dialogue system using typed sentences.

1. Introduction

The recent advance of speech science and related technology has made it possible to build continuous speech recognition systems working in real time. Using such systems as an interface, we can construct man-machine dialogue systems [1],[2]. In the speech dialogue system, the discourse analysis, that is, the analysis of structures of dialogues plays important roles in interpreting utterances.

This paper describes a method for the discourse analysis performed in the speech dialogue system we are developing [3]. The purpose of the discourse analysis is to provide the system with top-down hypotheses on words likely to appear in utterances of the partner of a dialogue.

The task performed through the man-machine dialogue is to make plans, for example, plans for seeing the sights of a city. The system is supposed to have a relational database about the sights of the city. A user (speaker) can access to the system by voice and collect information necessary to make plans.

The speech dialogue system consists of three components; a speech interface, a dialogue controller and a planner. The speech interface recognizes utterances from users and passes their semantic interpretations to the dialogue controller. The dialogue controller analyzes structures of conversations and returns to the speech interface top-down expectations of what would be said next. The planner gives to the controller a guide line on how it should carry out a conversation with users as well as helps users to make plans by supplying information requested and suggesting some tour plans.

The discourse analysis is performed in terms of topics and discourse goals based on [4]. The transition of topics through a conversation is represented by an AND-OR tree. In this formulation the prediction of topics likely to be mentioned in the next utterance is equivalent to

the determination of which node of the AND-OR tree is to be expanded next.

Each utterance of a dialogue has its own purpose (speaker's intention). The purpose of an utterance is called discourse goal. Since successive utterances are grouped to achieve a larger goal, discourse goals of a conversation can be described by a context free grammar, which we call discourse grammar. The top-down analysis of a sequence of utterances (a partial dialogue) by this grammar predicts discourse goals the next utterances could express.

A simulation of the dialogue system using typed inputs has proved that these top-down hypotheses can reduce a search space in recognizing utterances by about 60%.

- U101 I'd like to do a few day sights in Kyoto.
 S101 Which are you interested in, temples, gardens, or traditional artifact?
 U102 I've heard Kyoto is famous for gardens.
 S102 I see. I introduce two day tour for visiting gardens. The gardens in what eras would you like to visit?
 U103 Muromachi, Momoyama and Edo.
 S103 O.K. Famous gardens in Muromachi era are the stone garden of Ryoanji temple, Kinkakuji temple and Ginkakuji temple.
 U104 What is the stone garden?
 S104 It is made of a few large stones and white sand only. It is believed that the simplicity of the garden is related to "Zen".
 U105 That's great! Who built it?
 S105 I'm sorry, it is not known.
 U106 Who built Ryoanji?
 S106 It's believed that Katsumoto Hosokawa did.

Fig. 1 An example of the dialogue.

2. Representation of the context

Fig. 1 illustrates an example of dialogues which

the system would have with users. This example shows the conversation is developing in two different modes; system-initiative and user-initiative modes. In the system-initiative mode, the system, repeating questions to the user, elucidates the specification of his sightseeing tour, that is, a period of the tour, hotels to stay, places to visit and so forth, and then offers some candidates for these items. In the user-initiative mode, the user, inquiring detail information on these items, decides what are worth to involving in his plan. The questions issued by the user in this mode are interpreted as retrieval commands to the relational database. Responses of the system are generated based on the retrieved information.

2.1 Topic transition tree

It has been known [5] that topics in a goal-oriented dialogue move according to a task-dependent tree structure. In fact the topics in the illustrated example are specialized along the structure as shown in Fig. 2, which we call the topic tree. Nodes of the topic tree are entities related to the database such as names of relational tables, items included in the tables, and values of items. These correspond to topics the system can understand.

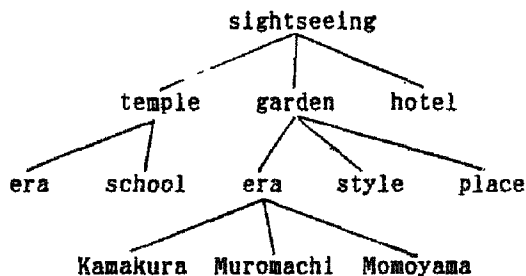


Fig. 2 A part of topic tree.

As we have reported in [3], however, an AND-OR tree is more suited for representing movements of topics than a simple tree. In the AND-OR tree, which we call a topic transition tree, AND-nodes represent topics introduced by the user, and OR-nodes represent topics introduced by the system. If the user inquires about two or more sights (each assumed to be a topic), the system must offer information on all of them. On the other hand, even if the system proposes two or more candidates for a visit, the user is not interested in all of them, but might move to the other topics. An AND-OR tree is suited to reflect this difference. The topic transition tree is considered a trace of a subtree of the topic tree.

2.2 Discourse goals

Each utterance in a conversation has its own purpose (speaker's intention). In the dialogue

illustrated in Fig. 1, an utterance U101 requires to make plans for sightseeing and presents information on a period of the tour. The following four utterances S102 to U103 have a larger purpose that the system tries to know what places the user wants to visit. The next utterance S104 proposes candidate sights to visit. In the utterances following S104, the user asks questions on those sights to judge which of them are worth to visiting, and the system provides some information on them. Thus discourse goals in a dialogue form a hierarchical structure like a tree.

This hierarchical discourse structure can be described by the context free grammar in which terminal symbols are discourse goals corresponding to a single utterance and nonterminal symbols are larger discourse goals corresponding to a group of utterances. Fig. 3 shows an example of the grammar for the discourse analysis. Underlined strings indicate terminal symbols. For example, a terminal symbol 'prst-alt' (present alternatives) represents a discourse goal of an utterance, like S102 in Fig. 1, used to present multiple choices, and a terminal symbol 'slect-alt' (select alternatives) represents a discourse goal to select one of alternatives presented. 'rqst-spec' (request for a part of specification) is a discourse goal of wh-questions issued by the system, and 'ans-spec' (answer a part of specification) works as an answer to 'rqst-spec'.

An utterance could have different discourse goals in different contexts, and a discourse goal could be expressed by various forms of utterances. Thus the relation between utterances and discourse goals is many to many correspondence.

- | | | |
|----------------|-----|-------------------------------|
| (1) mk-plan | --> | exm-spec, exm-plan |
| (2) exm-spec | --> | exm-spec, exm-spec |
| (3) dcd-spec | --> | dcd-spec, exm-spec dcd-spec |
| (4) exm-plan | --> | exm-plan, exm-plan |
| (5) exm-plan-1 | --> | exm-plan-1, exm-plan-1 |
| (6) dcd-cand | --> | (aqr-knwdg)*{, response} |
| (7) aqr-knwdg | --> | rqst-knwdg, ans-knwdg |
| (8) response | --> | accept reject |

Fig. 3 A subset of rules for the discourse analysis.

{ } indicates an optional term, and (x)* indicates a null string or the repetition of x as many times as necessary. Rule (2-3), for example, refers to the second rule with the third alternative of the right hand side.

3. Analysis of discourse structure

Fig. 4 shows the flow of the discourse analysis. It involves bottom-up and top-down analyses. The bottom-up analysis performs the semantic analysis of an input utterance and then makes the bottom-up hypotheses, that is, candidates for topics and discourse goals of the utterance. The top-down analysis predicts topics and discourse goals likely to appear in the current utterance referring to the context so far restored which is represented by the AND-OR tree of topics and the parsing history of discourse goals. The bottom-up hypotheses are matched against the top-down predictions. The best match gives the interpretation of the utterance, which is preserved as contextual information.

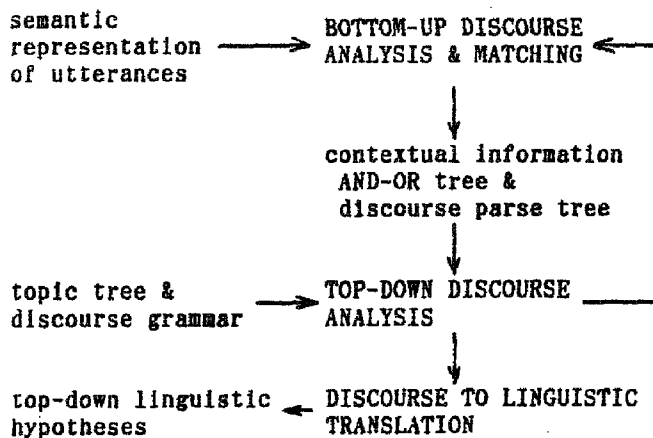


Fig. 4 Flow of the discourse analysis.

3.1 Bottom-up Analysis

The first stage of the bottom-up analysis is the semantic analysis of utterances, while the syntactic analysis of them is supposed to finish in the speech grammar. It is performed based on the case grammar. Case frames associated with verbs are used to represent the meaning of sentences in the case grammar. They are described by a set of slots, each indicating one of relations between the verb and a noun phrase, like an agent, object and instrument. The semantic analysis assigns noun phrases included in an utterance to some slot of the case frame of the main verb based on semantic markers of the noun phrases. The semantic interpretation of an utterance is represented by a list of four terms, a main verb, a case frame with slots filled, aspect information and the style of a sentence.

The head nouns of the case slot fillers (noun phrases) are proposed as candidates for the topic of the utterance being analyzed. Those of cases of topic, object and purpose are given a higher priority than others.

A lexicon is prepared to make bottom-up hypotheses on discourse goals. It contains the relation among a verb, aspect, the style of a sentence including the verb, and a discourse goal which the verb could express. Bottom-up hypotheses on the discourse goal are built up by consulting this lexicon. It is generally difficult to uniquely determine the topic and discourse goal of an utterance only by the bottom-up analysis.

3.2 Top-down Analysis

As mentioned in section 2.2, a grammar for the discourse analysis is formulated by the context free grammar. Thus an analysis of the conversation so far carried out results in a tree structure. Leaves of a discourse parse tree correspond to utterances. Fig. 5 shows an example of the top-down discourse analysis. It illustrates a discourse parse tree resulting from utterances U101 and S102 shown in Fig. 1 and discourse goals possible to be expressed by an utterance following S102. Rules used in the analysis of utterances U101 and S102 are rules (1), (2-1), (2-3), (3-1), (2,2), ((3-2) or (3-3)). The last stage of the rule applications is ambiguous and incomplete. It is ambiguous in that there are two applicable rules (3-2) and (3-3), and incomplete in that all terminal symbols of the right hand sides of the applied rules are not consumed. Thus the next discourse goal would be 'slct-alt' if the rule (3-2) is applied and 'chng-spec' if the rule (3-3) is applied.

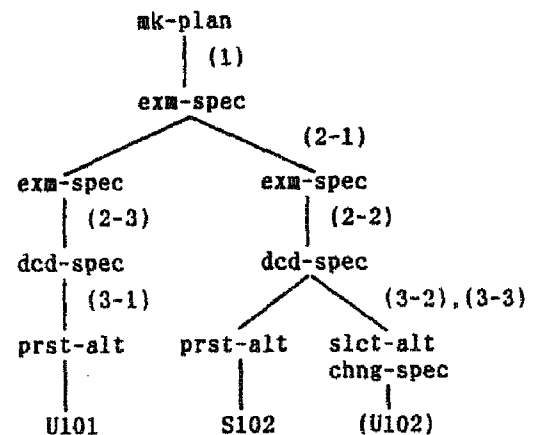


Fig. 5 An analysis of a discourse structure and a prediction of discourse goals.

The movement of topics through a conversation is stored in the topic transition tree. Thus for the top-down prediction of topics it is necessary to determine which node of this tree is to be selected or expanded next. New nodes which will result from the expansion can be known by referring to the topic tree because the topic transition tree is considered a trace of a sub-

tree of the topic tree. In this connection, the concept of a focus is very important. The focus means the topic currently focused. We define it as the last of topics which either of the system or the user has uniquely mentioned.

Which node is to be selected or expanded depends on the discourse goal of the next utterance. Fig. 6 shows the relation between discourse goals and topics in the top-down prediction. In order to shift the focus to nodes at higher levels than the currently focused node, it is necessary that all the topics under the new focus have terminated. The conditions for a topic to terminate can be stated as follows.

- (1) A node with AND successors can terminate only when all the successors have terminated.
- (2) A node with OR successors can terminate when one of the successors has terminated.

disc. goals	node to select or expand
slct-alt	select one of successors of the focus.
ans-spec	expand the focus.
chng-spec	expand a node at a higher level or select and expand one of successors of the focus.
rqst-cand	move the focus to a node at a higher level.
rqst-knwldg	expand the focus.

Fig. 6 The Relation between discourse goals and topics in the top-down prediction.

3.3 Translation of discourse hypotheses into linguistic hypotheses

A topic can be express by several words. For example, a topic 'garden' is expressed by two Japanese words 'niwa' (Japanese origin) and 'teien' (Chinese origin). So we have a table by which real words can be looked up from conceptual topic words. By using this table top-down hypotheses on the topic are translated into words likely to appear in utterances.

As mentioned in section 3.1, we have the lexicon describing the relation between a verb and discourse goals which the verb can express. This lexicon is also used to translate top-down hypotheses on the discourse goal into linguistic ones. First a set of verbs capable of expressing a hypothesized discourse goal is found by consulting this lexicon, and then semantic categories of those nouns which can occur together with these verbs are obtained by looking up case frames of these verbs.

4. Effect of the discourse analysis

The speech dialogue system reported here has not been complete. So we simulated the dialogue

system using typed sentences in order to measure an effect of the discourse analysis described in the previous sections. The dialogue controller accepted a dialogue consisting of typed sentences and generated top-down discourse hypotheses every time a sentence was input, and then the linguistic processor of the speech interface analyzed the sentence following the one just input in the dialogue, using the linguistic constraints translated from the discourse hypotheses and predicted words possible to follow each word of the analyzed sentence.

The average number of the predicted words, a kind of branching factor, was computed as a measure of the effect of the discourse analysis on the speech recognition. Assumed the vocabulary consist of about 600 words of which the nouns are about 360, a conversation composed of 60 sentences was analyzed. The average number of the predicted words was 240. This means that the discourse analysis has reduced the vocabulary size by 60 %.

5. Conclusion

The method for the discourse analysis performed in the speech dialogue system we are developing has been reported. The contextual information is analyzed in terms of topics and dialogue structures.

The discourse analysis involves the bottom-up and top-down analyses. The bottom-up analysis proposes candidates for topics and discourse goal based on the semantic representation of an utterance being analyzed. The top-down analysis makes hypotheses on topics and discourse goals referring to the contextual information. The bottom-up hypotheses are matched against the top-down hypotheses. The best match gives the interpretation of an utterance.

The top-down hypotheses are translated into the hypotheses at the linguistic levels, which are given to the speech recognition system. A simulation of the dialogue system using typed sentences has proved that these hypotheses can reduce a search space in recognizing utterances.

REFERENCES

- [1] Young, S.J. and Proctor, C.E., Computer Speech & Language, 3 (1989) pp. 329-353.
- [2] Young, S.R. et al., Com. ACM, 32 (1989) pp. 183-194.
- [3] Niimi, Y. and Kobayashi, Y., Preprints of the Second Symposium on Advanced Man-Machine Interface Through Spoken Language, (1988) pp. 33.1-33.8.
- [4] Grosz, J.B. and Sidner, C., Computational Linguistics, 12 (1986) pp. 175-204.
- [5] Grosz, J.B., Discourse knowledge, in: Walker D.E. (eds.) Understanding Spoken Language (North-Holland, New York, 1978) pp. 229-337.

SUSKIT-II—A Speech Understanding System Based on Robust Phone Spotting—

Yutaka KOBAYASHI†, Masanori OMOTE†, *Members*, Hidenori ENDO†, *Nonmember*
and Yasuhisa NIIMI†, *Member*

SUMMARY This paper describes an overview of our speech understanding system and reports on the recent results of the sentence recognition experiments. The system, we call SUSKIT-II, recognizes database queries in natural Japanese sentences. The user is expected to speak sentence by sentence. Among the difficult problems to overcome, this study paid the prime attentions to how to cope with the contextual variations of pronunciations and how to verify partial sentence hypotheses in a hierarchical system. The SUSKIT-II predicts words strings in a top-down manner, however, the verification of hypotheses against the input speech is done using a unit independent of word boundaries. Words are not suitable units of verification because the smoothing effect owing to phonetic contexts makes it difficult to recognize short words. In order to avoid the mis-recognition caused by the smoothing effect across word boundaries, the SUSKIT-II dynamically extracts those phoneme strings bounded by the easily detectable phonemes from the predicted word string as verification templates. The left-to-right time-synchronous beam-search strategy was adopted for searching likely sentences. We carried out sentence recognition experiments using the speech corpus consists of 159 sentences read by three Japanese male speakers. The task perplexity was 8.3. Using the speaker-dependent HMM parameters, we obtained the sentence recognition rates of 83.0–92.5%.

1. Introduction

Man-machine communication through speech would offer a great deal of convenience to our life. The intensive studies of speech understanding systems (SUSs) began in 1970s. The DARPA of the United States has been promoting development of the second generation SUSs since 1985 aiming at an ambitious target: a word accuracy of better than 95%, speaker-independent, continuous speech, 10,000 word vocabulary with natural grammar having a perplexity of 100 to 200, and with moderate noise and low stress. The European projects have diverse goals from practical isolated word recognition to speech dialogue systems⁽¹⁾. In Japan, the ATR Automatic Interpretation Telephony Laboratories was established in 1987. Their short term goal is a construction of a speaker-dependent or speaker-adaptive speech recognition sys-

tem. They assume cooperative speakers, good quality speech and pauses between syntactic phrases⁽²⁾. On the other hand, many laboratories have been engaged in the three-year project "Advanced Man-Machine Interface Through Spoken Language" started in 1987. The contractors of the SUS group including us set out the target of improving the medium-size vocabulary system and refining the use of the higher linguistic knowledge sources⁽¹⁾.

These several years, we have been developing a speech understanding system, SUSKIT-II, an abbreviation for the Speech Understanding System of Kyoto Institute of Technology, Version 2^{(3)–(5)}. We adopted the hierarchical architecture because we expected to separate the knowledge sources from the control. Unlike the black-board model, the pieces of knowledge themselves do not include concept of activating other pieces of knowledge. The connections among the component modules are fixed. The network model was not adopted either because we thought the hierarchical model is more tractable as we are interested in extending the current system to a speech dialogue system where the semantics, the pragmatics and the dialogue knowledge including the user model would play more important roles to decode the input speech.

The most specific feature of the SUSKIT-II is the choice of the matching unit. The task syntax and semantics of an SUS are usually described in terms of words, but speech wave suffers from heavy smoothing effect due to the physical constrictions. We considered the word-by-word matching is not suitable because the smoothing effect owing to the phonetic context makes it difficult to recognize short words. Note that in Japanese language we have a lot of short functional words which play important roles to indicate syntactic and semantic constituents. In order to avoid the mis-recognition caused by the smoothing effect across word boundaries, we introduced a concept of *robust phones*—those phoneme classes which are easily detectable. A verification template in the SUSKIT-II is a portion bounded by the robust phones extracted from a connected template of the predicted word strings. We call such a portion a *bounded phoneme string* (BPS) although it has a lattice form in general. Accordingly the system has another processing level below the word

Manuscript received January 10, 1991.

Manuscript revised April 8, 1991.

† The authors are with the Faculty of Engineering and Design, Kyoto Institute of Technology, Kyoto-shi, 606 Japan.

level in the hierarchical architecture where a node of the search tree corresponds to a BPS. Thus, the verification of hypotheses against the input speech was done using a unit independent of word boundaries. We expected the tree searching on the sub-word level to relieve greatly the problem of coarticulation across word boundaries and the mis-recognition of short functional words. Our approach can be considered as a combination of a hierarchical and a network architectures.

We selected the class of unvoiced fricatives, the closure before unvoiced stops and occasional pauses between words as the robust phones. The strident unvoiced fricatives /s/, /sh/, /ts/ and /ch/ and short silence intervals before unvoiced stops /p/, /t/ and /k/ can be detected with high accuracy in our language. Discrimination among /p/, /t/ and /k/ by machine is difficult as in most languages but in Japanese a short silence interval quite often precedes them, so such a short silence or a deep dip of the short-term energy is detectable as a good cue of a stop consonant. The SUSKIT-II first locates the robust phone candidates in the input speech. Then the matcher compares the BPS against the candidate intervals.

This paper describes the recent results of sentence recognition experiments. The matching module based on the HMMs for phoneme classes was used unlike the lattice matching version we reported earlier. The influence of the weighting functions of the weighted cepstral distance was also studied through the sentence recognition experiments.

Following a system overview, the control strategy is explained using a concrete example in the next two sections. Section 4 describes how to find robust phone candidates. And the remaining sections explain about the sentence recognition experiments.

2. Overview of SUSKIT-II

We have a small relational database containing information relevant to sightseeing in a town—temples, shrines, museums, hotels, transportations, and so on. The SUSKIT-II is a speech interface to the question answering system to the database. The user can ask questions in a restricted Japanese language spoken sentence by sentence. Sample queries are shown in Fig. 1.

The system consists of five major modules as shown in Fig. 2—an acoustic processor, a linguistic processor, a phonological component, a matcher and a controller.

The *acoustic processor* samples the input speech at 10 kHz and digitizes with the 12 bit precision. Then, it performs signal processing based on the linear predictive coding (LPC). For each 10 ms of speech the processor calculates the LPC-cepstral coefficients (LPC-CEPs), their time-derivatives (DELTA-CEPs),

Kousendai no denwabangou o oshiete kudasai.
(May I have the telephone number of Kousendai?)

Kinkakuji no haikanryo wa hyakuen desuka?
(Is the entrance fee of Kinkakuji temple 100 yen?)

Nyuujouryou ga nihyakuen ika de juuji yori
hayaku hiraku bijutukan o oshiete kudasai.
(Are there any museums which open before 10 o'clock and
whose entrance fee is less than or equal to 200 yen?)

Fig. 1 Examples of acceptable sentences.

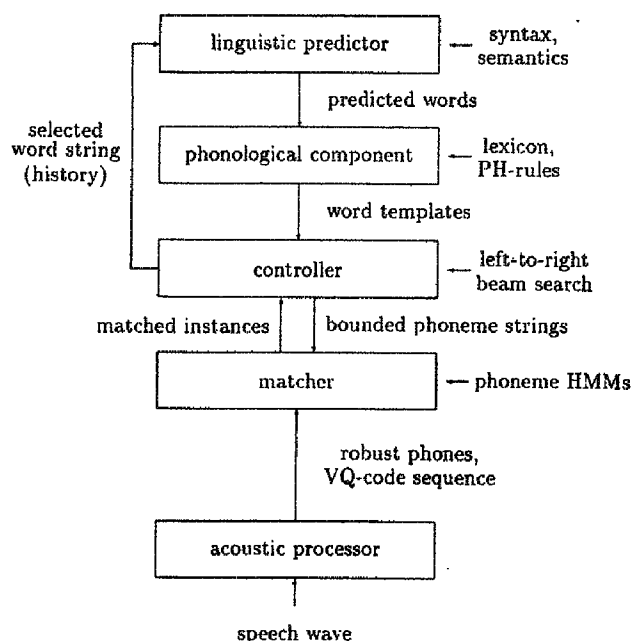


Fig. 2 An overview of the SUSKIT-II.

and a pair of the short-term energy and its time-derivative (POWs). The time-derivatives are calculated as the differences of the LPC-CEPs and the POWs between the preceding and the following frames. Using a set of training data we designed separate codebooks for these three sets of parameters and obtained the parameters of three-state discrete Hidden Markov Models (HMMs) for 18 phoneme classes. The codebook sizes were 256. We grouped Japanese sounds into 18 phoneme classes as shown in Table 1. In order to cope with fast speech, the 18-th phoneme class was assigned a label *X*, which represents non-silence energy dips preceding unvoiced stop consonants. The acoustic processor outputs a sequence of vector codes, and the positions and the labels of the robust phone candidates in the input speech.

The *linguistic processor* generates partial sentence hypotheses as top-down predictions. The syntax and semantics of the task were described in definite clause grammar (DCG)—context free grammar augmented by semantic markers and compatibility check predi-

Table 1 Phoneme classes.

vowels		consonants	
A	a	N	m n ng
I	i	Z	z dz
U	u	R	r
E	e	D	b d
O	o	G	g
semivowels		S	s sh ts ch
Y	j	H	h
W	w	K	k
other		P	p t
Q	silence		
X	dip		

cates in our case. Given a task grammar in DCG, we developed two algorithms for deriving the linguistic processor programs in Prolog which work as an efficient parser/predictor especially for an SUS; one for the left-to-right control⁽⁶⁾ and the other for the island-driven control⁽⁷⁾. We implemented the former in the SUSKIT-II. The vocabulary size was 248 and the task perplexity was 8.3⁽⁸⁾.

The *phonological component* deals with modifications of pronunciation template in order to reflect the smoothing effect caused by phonetic contexts. The variations inside a word are precompiled off-line into a lattice style like a phoneme lattice. We construct the *lexicon*—a set of word templates—from Japanese orthographic word transcriptions of the task vocabulary by applying the standard duration formula and the phonological rules. Inter-word variations are taken into account during the sentence recognition process when adding a word to an existing partial sentence hypothesis in order to make a longer one. The phonological component marks robust phones in the template of the word string, and divides the template there.

The *matcher* calculates the likelihood of the sequence of predicted words by comparing it with the input speech. Exactly speaking, the matching proceeds from left to right using the BPSs as matching units. Given a BPS and a starting frame in the input speech, the matcher first constructs a network of the phoneme HMMs according to the BPS which is generally in a lattice form. Then, it looks for the robust phone candidates in the input within twice the standard duration of the BPS, starting from the current frame. After deciding a few candidate intervals which end up with the same robust phone as that of the BPS, the matcher computes the probability that the network generates the sequence of vector codes obtained for each interval. The negative of the normalized log-probability is returned to the controller as the distance between the BPS and the particular interval.

The *controller* invokes the other components of the system so that they work in harmony. Particularly, it holds partial sentence hypotheses, selects some of them at a time for verification or for expansion, and discards unlikely candidates. We use three levels of

search trees for the benefit of sharing information among several hypotheses. Verified hypotheses are chosen time-synchronously based on the beam search strategy.

3. Control Strategy

The controller keeps the history of partial sentence hypotheses of the different processing levels in three trees—the *syntactic category* (SC) tree, the *word* (WD) tree and the *phoneme* (PH) tree as illustrated in Fig. 3. The three trees are mutually linked for sharing information with each other level.

Speaking of the upper-most search tree (SC-tree), a path from the root to an arbitrary node corresponds to a sequence of syntactic categories. Since we describe the linguistic constraints from the syntax and the semantics of the task in a unified framework of the DCG, we use a *syntactic category* (SC) in the sense of the name of a word group which are considered equivalent by the linguistic processor. The linguistic processor predicts syntactic categories but not individual words. The controller retrieves the predicted words by consulting the dictionary.

The second tree (WD-tree) represents word-level hypotheses. We associate a word with each node, then the word sequence along a path from the root to any node makes up a partial sentence hypothesis represented in terms of words. Each node has one or more links to some SC-nodes. Usually those words in the same SC share an SC-node. On the contrary, since it is possible for a certain word to be predicted from more than one SCs, we let them share the word in order to reduce the number of alternatives to be verified by the matcher.

The lowest search tree (PH-tree) reflects pronunciations of all the partial sentence hypotheses predicted so far. Inter-word variations are taken into account later when the phonological module appends the pronunciation templates of likely words following the hypothesis in attention to their predecessor. A BPS is attached to a PH-node. A path from the root to any node corresponds to a typical pronunciation of a partial sentence hypothesis up to some robust phone. Those BPSs associated with non-leaf nodes are ready to be compared against the input. However, those associated with leaves of the PH-tree may not necessarily end with a robust phone. They must be extended by concatenating the templates of the possible following words before comparison so that the templates would end with a robust phone.

Figure 3 shows the situation the word sequences /chorakuji no soba/ and /chorakuji no haikanryo/ were decomposed into three BPSs /choraQ/, /kujinos/ and /oba/, and /choraQ/, /kujinohaiQ/ and /kanryo/, respectively.

A matching result is represented by a quadruple:

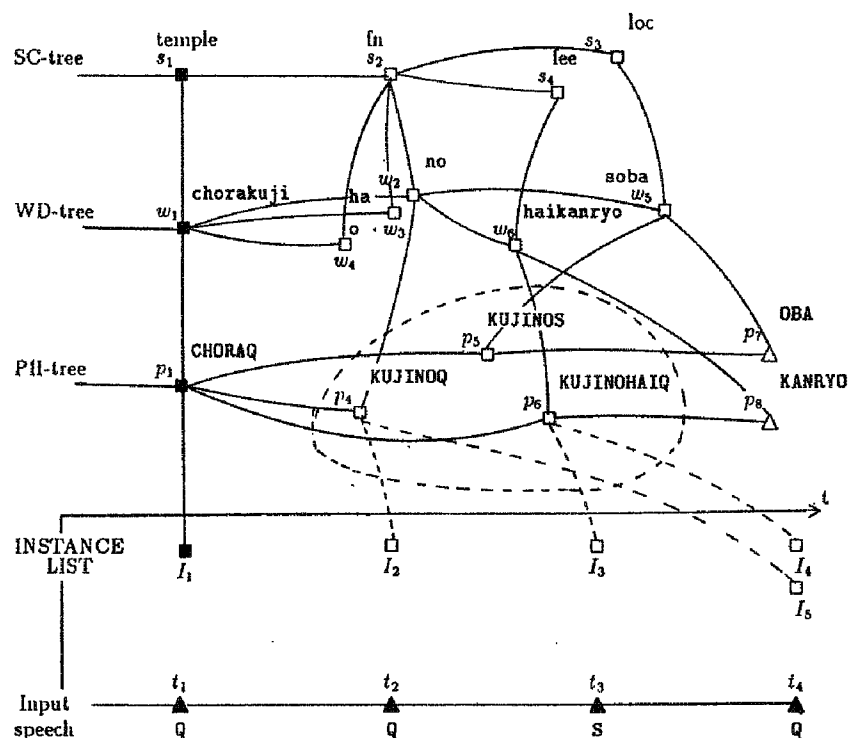


Fig. 3 An example of the hierarchy of the three search trees for /choraku ji no haikanryo.../.

a PH-node, an accumulated verification score, a start time and an end time of matching. The instances of matching results are registered in a two-dimensional array spanned by time and score axes. We call the array the *instance list*. The horizontal axis is the time axis. Instances are stored at their end time along the vertical axis in the increasing order of the verification score of the matching result.

There are three types of nodes in the PH-tree. Leaf nodes are indicated by white triangles. The black boxes stand for nodes already matched (matched nodes). The white ones represent nodes whose BPSs are ready to be compared against the input speech. It should be noted that more than one white boxes appear before a leaf node. Those nodes which directly follow matched nodes are placed in a queue for matching and expansion. The black triangles along the input represent the positions of the robust phones detected by the bottom-up analysis. Since the matching units are bounded by robust phones, the instances are stored basically at these positions in the *instance list*. The dashed lines indicate that the matching results for p_4 , p_5 and p_6 will be stored at time t_2 , t_3 and t_4 .

As we adopted the time-synchronous beam-search strategy for finding likely sentences, the number of instances at a particular time is restricted below a certain threshold. When an instance was selected, the corresponding PH-node comes into attention. The controller selects all the instances from time t_1 before selecting any from time t_2 where $t_1 < t_2$.

4. Spotting Robust Phones

The matcher determines the interval of interest from the input speech by referring to the locations of robust phones. Therefore, the robust phones must be detected before the linguistic decoding.

First, we expected the following two sound groups as the robust phones: (1) the unvoiced fricative group S which includes /s/, /sh/, /ts/ and /ch/, and (2) the silence Q which would precede unvoiced stops /p/, /t/ and /k/. The robust phones were detected by combining the absolute and the relative spotting results. The absolute spotting refers to finding intervals where the probability that the HMM for the corresponding phoneme class generates the observed sequence of VQ codes exceeds a certain threshold. On the other hand, the robust phones found by the phoneme discrimination are considered relatively more probable than other phoneme classes for the particular intervals. We first removed the candidates having less than three frames from consideration, and then merged these results by applying the following criteria:

- (1) The overlapping Q -intervals detected by the two different spotting criteria are merged into one.
- (2) For the unvoiced fricative S , if S -intervals appears in the absolutely spotted result within a relatively-spotted S -interval, the latter interval is labeled as S .

The spotting results of the robust phones for the 17 phoneme case are shown in the upper half of Table

Table 2 The spotting results of the robust phones.

speaker		KB	NY	HA	average
17 phonemes	correct(%)	98.0	98.6	96.0	97.5
	deletion(%)	2.0	1.4	4.0	2.5
	insertion(%)	6.1	4.3	16.9	9.1
18 phonemes	correct(%)	98.4	99.3	98.3	98.7
	deletion(%)	1.6	0.7	1.7	1.0
	insertion(%)	6.6	6.7	7.4	6.9

2. The detection rate for the fast-speaking speaker HA was considerably lower than those for the other two, and the many false alarms were detected.

Next, we introduced another phoneme group X corresponding to the deep energy dips and added X to the robust phones, because we often observed X s before unvoiced stops instead of Q s in the speech corpus of speaker HA. For other speakers, there were not enough number of X s in the training data, so the X s were fed to the HMM for Q . In this case, we considered Q s and X s as equivalent in the sense that they both signals the existence of an unvoiced stop.

The results of spotting the robust phones for the 18 phoneme case are shown in the lower half of Table 2. One can see the detection rate for speaker HA was improved by about five percents, and the false alarms were reduced largely.

5. Conditions of the Experiments

The speech corpus used in a series of the sentence recognition experiments consists of 53 sentences read by each of three Japanese adult male speakers, namely, KB, NY and HA. The corpus contains about 9.4 minutes of speech in total.

For each speaker, we used eleven sentences from the first half in order to design the codebooks. The HMM parameters obtained from the initial half of the speech corpus of a speaker were used to recognize his latter half, and vice versa. We trained the HMM parameters of phoneme classes using the forward-backward algorithm. In Experiment (1) the 17 phoneme classes except the dip class X were distinguished, while Experiments (2)-(4) assumed 18 phoneme classes.

The weighted cepstral distance measure was used in the codebook design and the vector quantization, where the weights were

$$w_n = 1 + M \sin \frac{\pi n}{N} \quad (n = 1, \dots, N). \quad (1)$$

$N = 14$ and we carried out recognition experiments for $M = 0$ (Experiment (4)), 3 (Experiment (3)) and 7 (Experiments (1) and (2)). $M = N/2 = 7$ corresponds to the weighting function proposed by Juang et al.⁽⁹⁾

Table 3 The sentence recognition rates.

Experiment	Phoneme classes	Weight factor(M)	speaker	Correct(%)	
				first	~fifth
(1)	17	7	KB	88.7	90.6
			NY	86.8	90.6
			HA	58.5	67.9
			Ave.	78.0	83.0
(2)	18	7	KB	90.6	92.5
			NY	92.5	98.1
			HA	83.0	86.8
			Ave.	88.7	92.5
(3)	18	3	KB	86.8	92.5
			NY	90.6	94.3
			HA	77.4	86.8
			Ave.	84.9	91.2
(4)	18	0	KB	90.6	90.6
			NY	90.6	90.6
			HA	81.1	86.8
			Ave.	87.4	89.3

Table 4 Phoneme recognition rates.

speaker	KB	NY	HA	average
correct(%)	86.8	91.8	85.1	87.9
substitution(%)	9.4	5.7	8.6	7.9
deletion(%)	3.8	2.5	6.3	4.2
insertion(%)	7.7	9.2	10.9	9.3
speaking rate (morae per sec.)	6.6	6.1	7.2	6.6

6. Experimental Results

Table 3 shows the sentence recognition rates measured in terms of percentage understood out of 53 sentences. We considered a sentence was understood if the recognized word string is translated into the intended database query, though we do not explain the translation module here. The columns, 'first' and '~fifth', stand for the rates of the correct sentences out of the 53 sentences in the first position and within the top five, respectively. The inter-speaker variability was not small.

The results of the phoneme-class discrimination are obtained while the acoustic processor is spotting robust phones. Table 4 summarizes the phoneme-class recognition accuracies for the three speakers. We obtained the average phoneme-class recognition rate of 87.9%. Speaker HA had the fastest speaking rate of 7.2 morae per second, occasionally around 9 morae per second. A mora is a rhythmic unit usually consisting of a consonant followed by a vowel in Japanese. His deletion errors were about twice as many as other speaker's. In general, the deletion errors considerably decrease the performance of the continuous speech recognition.

The sentence recognition rates reflect the phoneme recognition rates. Speaker HA spoke considerably faster than the other two. Accordingly his speech suffered heavily from the smoothing effect. Twice as many phoneme deletion errors for speaker HA made the sentence recognition more difficult than for the

other speakers. Especially, the detection rate of the robust phones was only 96.0% for him. We found in many cases the silence intervals before unvoiced stop consonants were reduced to an energy dip. It was the reason why we added such a dip as a new phoneme class in the rest of experiments.

Speaking of Experiment (1), the error analysis taught us that the phonological rules, which expected a silence before the affricates /ch/ and /ts/, was not always appropriate. We let such a silence optional in the later experiments. We had inhibited the inter-word pauses when a content word is followed by a functional word. This word-juncture rule originally produced superfluous PH-hypotheses in experiment (1) if an unvoiced stop or an unvoiced fricative follows an inter-word pause. Both phonetic and syntactic contexts were taken into account in the new version, so that the controller could prune unlikely hypotheses more effectively. Additionally, in order to cope with the case where no robust phone is found within the scope determined from the standard duration of the BPS, the matcher was modified so that it returns the positions and the values of the local minima of the normalized distance between the BPS and the input speech. These matching results were also registered in the instance list.

In Experiment (2), after these modification, we observed the sentence recognition rates improved 1.9, 5.7 and 24.5% for speakers KB, NY and HA, respectively. We can conclude that the introduction of the dip phoneme class was really effective for speaker HA. For KB and NY, the recognition rates exceeded 90%, while 83% for HA. We consider the current set of the phonological rules might be insufficient to cope with the fast speech.

The sentence recognition rates for the non-weighted metric (Experiment (4)) were 1.9% lower for NY and HA than in Experiment (2). The results for the modest weighting function $M = 3$ (Experiment (3)) were a little worse than for the non-weighted metric.

7. Discussion and Conclusion

In this study, the SUSKIT-II dealt with a medium-size-vocabulary speaker-dependent continuous-speech recognition. The word-asynchronous matching strategy was effective as we observed no serious mis-recognition of short functional words. As we saw in the preceding sections, a proper choice of the weighted metric was important, though we examined only a few of variations. Since the average sentence recognition rate was 88.7% and the average sentence length was 6.7 words, the word accuracy can be estimated as $0.887^{1/6.7} \times 100 = 98.2\%$. The SPOJUS of Toyohashi University of Technology⁽¹⁰⁾ and the conference registration task of the ATR⁽¹¹⁾ had

similar vocabulary sizes and the task perplexities. The former system recognized about 60% of the input sentences. The latter demonstrated sentence recognition rate of about 85% for the speaker-dependent sentences spoken phrase by phrase. Although the direct comparison of the performance with other systems was difficult, the SUSKIT-II achieved the acceptable performance as a test bed for the study on the speech dialogue which we are now carrying out.

The average process time spent in the sentence recognition experiment was about one hour per a sentence on SUN Sparc Station 1 (12 MIPS, 8 MB memory). It consumed 98% of the SUSKIT-II's process time. We are now trying a few approaches to the reduction of the process time. One is based on the combination of the smoothed likelihood scores calculated from the phoneme HMMs and the dynamic time-warping. Another is based on a new block calculation method for the HMM trellis. The tentative experiments achieved several times faster processing speed without considerable degradation of the accuracy. Another choice is a matcher based on the symbol comparison. Assuming the acoustic processor output a phoneme lattice, such a matcher would be convenient when one can expect the phoneme recognition rate to be high.

Several mis-recognitions occurred at long compound words, such as /umekojijokikikanshakan/ (/umekoji/, /joki/, /kikansha/, /kan/). The current task grammar allows to consider the values in the database fields as single words. Therefore, the phonological component does not insert optional pauses between component words in such a long compound word. However, the user of the system might have a break within a long word. This problem should be attacked by studying the prosody and refining the task grammar.

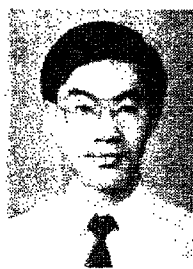
The SUSKIT-II predicts words strings in a top-down manner and the verification proceeds irrelevantly to the word boundaries. This is convenient for recognizing natural languages which have many short functional words as Japanese. The experimental results using speaker-dependent parameters showed sentence recognition rates of 83.0–92.5% for three speakers. Future problems for improvement of the system include (1) refinement of the task grammar, (2) enlargement of the vocabulary, (3) study of the speaker-specific phonological rules, and (4) speaker adaptation.

Acknowledgement

This study was partly supported by Grant-in-Aid for Scientific Research on Priority Areas "Advanced Man-Machine Interface Through Spoken Language".

References

- (1) Notes of Special Session: "Speech Processing in Human-Machine Interaction—An International View", ASA-ASJ Second Joint Meeting, Hawaii (Nov. 1988).
- (2) Shikano K. and Kurematsu A.: "Trends on speech understanding researches", *J. Acoust. Soc. Jpn.*, **42**, 12, pp. 948-952 (Dec. 1986).
- (3) Kobayashi Y. and Niimi Y.: "A new architecture of a speech understanding system—a hybrid of a hierarchical and a network models", In *Proc. Int. Conf. Acoust., Speech & Signal Process.*, pp. 1585-1588 (1986).
- (4) Kobayashi Y. and Niimi Y.: "An overview of linguistic decoding for a voice-input question/answering system", *J. Inst. Electron. & Telecom Eng.*, **34**, 1, pp. 96-101 (1988).
- (5) Kobayashi Y. and Niimi Y.: "A sub-word level matching strategy in a speech understanding system", *J. Acoust. Soc. Jpn. (E)*, **E9**, 3, pp. 137-145 (1988).
- (6) Niimi Y., Kobayashi Y. and Uzuhashi S.: "Top-down use of linguistic constraints in a speech understanding system", *Trans. IEICE*, **J70-D**, 9, pp. 1772-1782 (Sept. 1987).
- (7) Kobayashi Y., Niimi Y. and Uzuhashi S.: "Linguistic processing in an island-driven speech understanding system", *Trans. IEICE*, **J71-D**, 12, pp. 2563-2570 (Dec. 1988).
- (8) Kobayashi Y. and Niimi Y.: "Evaluation of a speech understanding system—SUSKIT-2", In *Proc. Int. Conf. Spoken Lang. Process.*, pp. 725-728 (1990).
- (9) Juang B. H., Rabiner L. R. and Wilpon J. G.: "On the use of bandpass lifting in speech recognition", In *Proc. Int. Conf. Acoust., Speech & Signal Process.*, pp. 765-768 (April 1986).
- (10) Hirata Y., Murase I., Tanoue T. and Nakagawa S.: "Evaluation of syntax/semantics oriented spoken Japanese understanding system — SPOJUS-SYNO/SEMO—", *IEICE Technical Report*, **SP89-96** (1989).
- (11) Kita K., Takezawa T., Hosaka J., Ehara T. and Morimoto T.: "Continuous speech recognition using two-level LR parsing", In *Conv. Rec. Acoust. Soc. Jpn.*, 3-8-17 (Sept. 1990).



Masanori Omote received the B. E. and the M. E. degrees from Kyoto Institute of Technology, Japan in 1989 and 1991, respectively.



Hidenori Endo received the B. E. and the M. E. degrees from Kyoto Institute of Technology, Japan in 1989 and 1991, respectively.



Yasuhisa Niimi received the B. E., the M. E. and the Ph. D. degrees from Kyoto University, Japan in 1962, 1964 and 1969, respectively. From 1964 to 1969 he was working at Kyoto University. Since 1970 he has been working at the Department of Electronics and Information Science of Kyoto Institute of Technology. Presently he is a professor and the Director of the Information Processing Center of the same university.

His current research interests include speech information processing, natural language processing and artificial intelligence. He published a book "Speech Recognition" in Japanese in 1979 and coauthored many books in the fields. Dr. Niimi is a member of the Information Processing Society of Japan, the Acoustical Society of Japan, the Japanese Society for Artificial Intelligence, and the EURASIP.



Yutaka Kobayashi received the B. E. and the M. E. degrees from Kyoto Institute of Technology, Japan in 1971 and 1973, respectively. He is an Instructor at the Department of Electronics and Information Science of the same university. His research interests include speech information processing, especially the development of speech understanding systems, and natural language processing.

Mr. Kobayashi is a member of the Information Processing Society of Japan, and the Acoustical Society of Japan, and the Japanese Society for Artificial Intelligence.

○ 北原 義典 天野 明雄 村山 秀樹 市川 薫
(日立・中研)

1. まえがき

柔軟な表現形態を持つ口語音声認識し、建築設計情報を検索するシステムを構築した¹⁾。限定されたタスクを扱ったQAシステムでは、列車座席予約システム²⁾や観光案内システム³⁾等の例がある。本システムは、語用論⁴⁾に基づく口語文生成ルールによって生成された概念語列よりなる文パターンを正規文法⁵⁾として持ち、さらに各概念語に、非注目語として類義語をグループ化した単語集合、及び注目語として相互に音響的距離が大である単語集合を割り当てることによって、深層構造を一意にとらえ表層構造の変化を吸収する。正規文法は、生成文法の中では最も制約の強い文法であるが、修正、追加等更新の容易性、処理の高速性等の利点があり、本検索システムのような限定タスク下で比較的シンプルな口語文を扱う条件のもとでは有効である。本稿では、本システムの口語文受理の方法について述べる。

2. システム構成

本システムは、建物の属性すなわち①用途②雰囲気③材質④色⑤設備⑥設計者及びその属性値で記述された建築物の空間デザインの写真データを検索するもので、ユーザの建物に対する希望を順次音声入力し(図1)、条件に合致する写真をカラー画像表示する。システムの構成を図2に示す。ガイド文に応答する形で入力された音声は分析部にて、ケプストラム、Δケプストラム、パワー+Δパワーを抽出、ベクトル

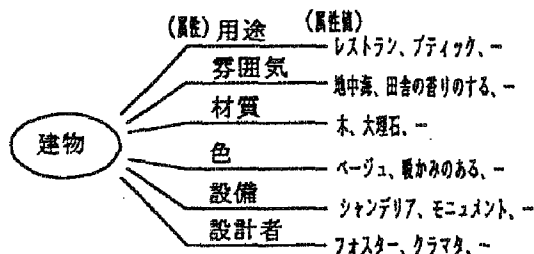


図1 建物の持つ属性と属性値

量子化された後、正規文法で記述されたHMMネットワーク上で照合が行われる。HMMはV CVを単位とし連結したものである。

3. 口語文の受理

まえがきでも述べたように、本システムでは、口語文受理正規文法(以下口語文法)及び類義語グループ単語辞書により口語文を処理する。口語文法は、口語文生成ルールにより検索用テンプレート文を口語文に変換した非注目概念スロット及び1~2個の注目概念スロットの列から構成される。各概念スロットは非注目語辞書、注目概念スロットは注目語辞書を各々保持し、HMM照合時に参照される。HMM照合は、文法で受理される文全てに対して行なわれ、各々尤度が算出される。「キャンセル」「検索開始」等のコマンドも柔軟な口語として入力できるよう検索文とともに口語文法として登録しておく。口語文法により受理される口語文は、概念スロット列の総数で一属性あたり平均367文、各

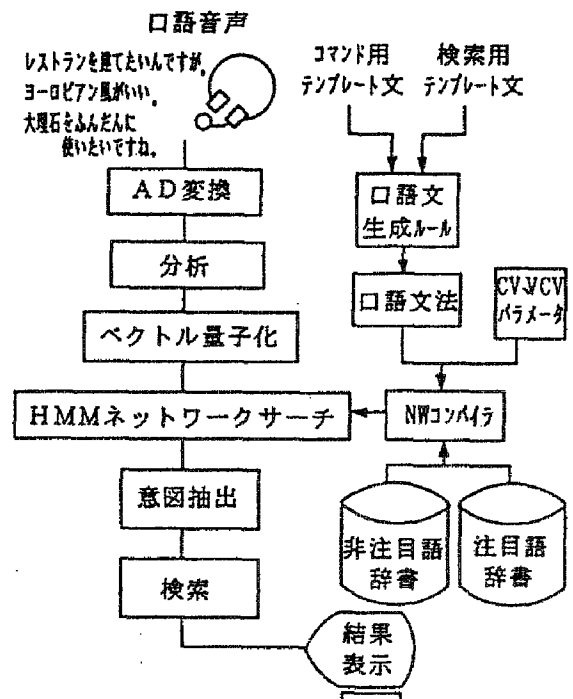


図2 システム構成

* Study on the grammar to accept colloquial speech in the information retrieval system

By Yoshinori Kitahara, Akio Amano, Hideki Murayama, Akira Ichikawa (HITACHI, Ltd)

概念スロットを非注目・注目語で置き換えた文の総数は同258718文であった。意図抽出部では、最尤系列に含まれる注目概念スロット（属性）内の単語（属性値）を同定することにより検索条件として検索部に送る。

3. 1 口語文生成ルール

口語文生成ルールは、図3に示す2段階の口語文特性に基づいて構成したもので、概念語列の書き換えルールである。同図に示す特性は、建物の6種の属性に関する質問に答える口語文（1文）で、ユーザの“希望”の表現を前提としており、7人の男女が発話した431文に基づいて整理したものである。同特性のうち、⑤『時制の変動』⑥『発話の中断』については例が少数であり、今回はルール化しない。本ルールの適用によって得られた口語文パターンにより“文型”の変動を吸収する。

3. 2 非注目語辞書

図3における③『語の置換』は類義語辞書に登録された単語を用いて行なわれる。本辞書により“語”の変動を吸収する。

3. 3 注目語辞書

検索の対象となる建物の属性の持つ値は、認

識結果が決定的なキーワードとなるため、語相互の音響的距離が大となるものを選択し、注目語辞書として登録しておく。語間の距離は、予め音韻認識を行ない、得られたコンフュージョンマトリクスをもとに、DP照合により算出したものである。

4. まとめ

口語音声を入力することによって、情報を検索するシステムの、柔軟な表現形態を吸収する方法について検討した。今後、本システムの評価を重ね、文法の精密化を図る予定である（部分評価については文献1）参照）。また、現在は、建築物のデータが属性及び属性値で記述されたテーブルによって検索を行なっているが、連想検索が可能なように、概念ネットワークによるデータの階層化を行ない、それに応じた口語文受理についても今後検討を行なう。

討論頂いた当研究所第6部藤沢主任研究員はじめ関係各諸氏に深謝致します。

- 1) 村山他：本予稿集（平3-3）
- 2) 好田他：音響学会誌 Vol. 34, No. 3 (昭53-3)
- 3) 新美他：信学論、Vol. J70-D, No. 9 (昭62-9)
- 4) 北原他：情処全大、3N-1 (昭60-9)
- 5) 村山他：音響講論、1-8-23 (平2-9)

- (1) 意図の移入
 <動詞連用><願望助動詞「たい」>
 <動詞未然><意志助動詞「よう」「う」>
 <動詞連体><意志を表わす名詞「つもり」「予定」>等

(2) 構文の変形

①語の付加

a. 複合辞
 [<判断を表わす動詞「と思う」>] [<断定助動詞「(ん)だ」「(ん)です」>]
 [<接続助詞「けれども」「が」>] [<親愛終助詞「よ」「ね」>] 等

b. 間投詞 「あー」「あの」「うーん」「えー」「ええと」等

c. 連用語 「やっぱり」「ちょっと」「絶対に」等

d. 接続詞 「実は」「どちらかといえば」等

e. 主題 「～としては」「～については」「～は」等

②語の省略

a. 旧情報、既出語 「レストランを建てたいと思う」⇒「レストランをと思う」等

b. 主語 「私はレストランを建てたい」⇒「レストランを建てたい」等

c. 場所格 「天井にシャンデリアが欲しい」⇒「シャンデリアが欲しい」等

③語の置換

a. 類義語 平易化：「建てる」⇒「欲しい」「作る」等
 短縮化：「けれども」⇒「けれど」「けど」等
 ぼかし：「が」⇒「なんか」「とか」「の方が」等

b. 照応 指示語：「建物の色」⇒「その色」等

④語順入替え

a. 場所格と対象格 「ピアノを客席に置きたい」⇒「客席にピアノを置きたい」等

⑤時制の変動

⑥発話の中断

図3 口語文の特性（検索文の場合）

1.はじめに

筆者等は、検索システム等への応用をめざして音声認識装置の開発を進めている。音声入力を用いた使い勝手のよいユーザーインターフェイスの構築には、自然な発声形態が望ましい。タスクを検索システムとし、入力を単文に限定することで正規文法を使用した。正規文法は、制約の強い文法ではあるが、平均分岐数を減少することによって認識性能を向上するために有効である。口語的な言い回しに対応できるように基本となる正規文法から規則に基づいて自動的に正規文法を口語向けに変換し、認識時の文法とする¹⁾。

認識は、正規文法から生成可能な文章をHMMのネットワークに展開し、ネットワーク上の最適パスをViterbiアルゴリズムによって求める。

今回、連続音声入力を適用した情報検索システムを構築し、検討を行なったので報告する。

2.システム概要

本システムの概要をFig.1に示す。

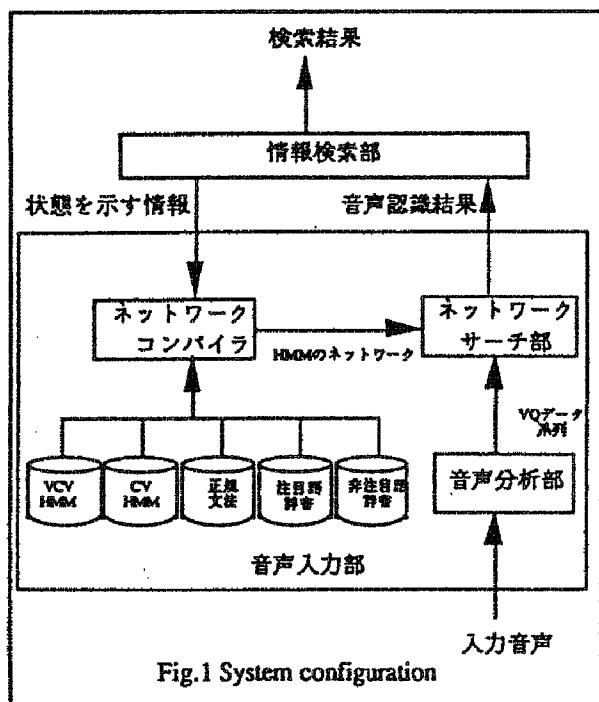


Fig.1 System configuration

本システムのタスクは、建築設計情報の検索である。検索の条件として用途や、雰囲気などの検索条件を入力し検索を行なう。システムは、情報検索部、音声入力部から構成される。情報検索部への入力は、音声入力部を通じて行なわれ、検索条件文及びコマンドの2種類に大別できる。検索条件文は、画面に表示された入力ガイド文にしたがって入力する。情報検索部の状態は、入力が進むにつれて、Fig.2に示す状態遷移図のように変化する。入力すべき検索条件文は、情報検索部がFig.2の状態遷移図上でどの状態にあるかによって一意に決定する。コマンドは、どの状態でも入力可能である。

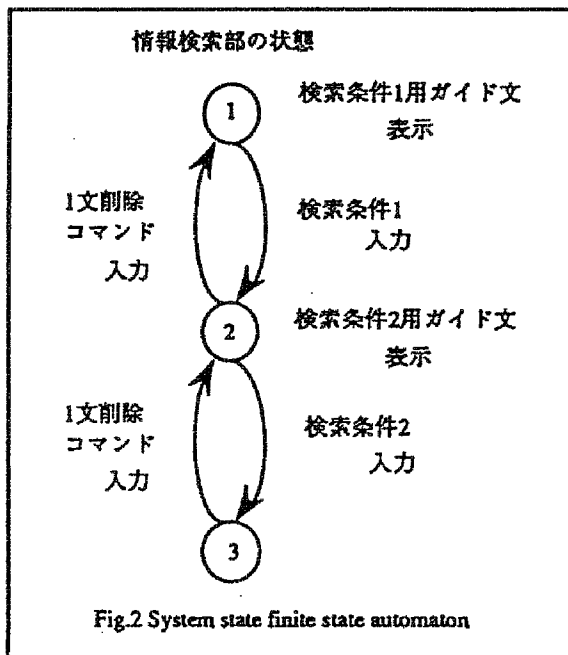


Fig.2 System state finite state automaton

音声入力部は、情報検索部がFig.2の状態遷移図の状態により、認識対象となる文を絞り込み、音声認識処理を行ない、情報検索部に出力する。

以下、音声入力部での処理の流れを示す。音声入力部は、音声分析部、ネットワークコンパイラ、ネットワークサーチ部に分れている。情報検索部がFig.2の状態遷移図の状態情報がネットワークコンパイラに渡される。ネットワークコンパイラでは、情報検索部の状態の情報により、文法、辞

*Information retrieval system using continuous speech recognition

By Hideki Murayama, Akio Amano, Yoshinori Kitahara, Akira Ichikawa (HITACHI, Ltd)

書、HMMパラメータから、HMMネットワークを作成する。音声分析部では、音声をA/D変換した後、分析を行ない、LPCケプストラム、 Δ ケプストラム、パワー+ Δ パワーの3つの特徴量のVQコードを出力する。ネットワークサーチ部では、音声分析部の出力したVQコードに対してネットワークコンパイラの出力したネットワーク上でサーチを行ない認識結果の文章を情報検索部に出力する。ハードウェア構成は、上記の情報検索部に汎用WSを一台使用し、ネットワークコンパイラ、ネットワークサーチ部には別の汎用のWS(8MIPS)を使用し、音声分析部には、高速信号処理を行なうArray Processorを使用している。Array Processorは、実時間で特徴量の分析を行なう。

3. 認識方式概要

本認識方式は、認識の基本単位としてCV及び、VCV音節を使用している。単語辞書は、音韻記号系列で記述されている。文法は正規文法を使用している。認識に先立ち、情報検索部の状態情報に基づき、文法、辞書の情報を使用して、文章のHMMを作成し、Viterbiアルゴリズムによるネットワークサーチを行なう。処理量削減のためにビームサーチを行なっている。

3.1. ネットワークコンパイラ

情報検索部からの状態情報により、HMMのネットワークを作成する。まず、情報検索部からの状態情報によって使用する正規文法を選択する。本システムで使用する正規文法では、終端記号が、特定の単語ではなく、単語の集合を示す概念に対応させている。ネットワークコンパイラは、概念に属する単語に展開し、更に単語辞書を使用してHMMのネットワークに展開する。この正規文法を表す有限状態オートマトンと単語を展開した後ネットワークをFig.3に示す。

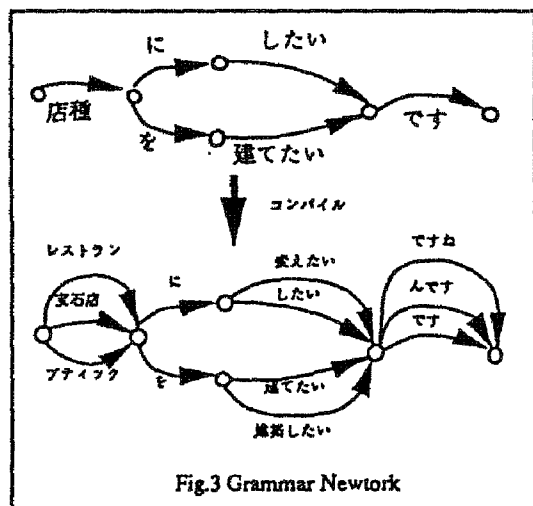


Fig.3 Grammar Network

単語辞書は、音韻記号系列の形で記述されている。単語間の接合部分は、処理量を削減するためにCVのHMMを、単語内部は調音結合に対応するためにVCVを使用した、HMMのネットワークを作成する。

3.2. ネットワークサーチ

Viterbiサーチを認識の基本としている。高速化を図るためにHMMの各状態を単位として枝刈りをするビームサーチ²⁾を行なっている。

4. 評価実験

評価実験としてコマンド及び検索条件文を示す文章に対する認識実験を行った。

4.1 実験条件

評価用データは、平均単語連結数5.4、平均分岐数3.6、状態数93、総アーク数337、生成可能文章数5430の有限状態オートマトンを文法として、使用した。

4.2 実験結果

女性1人のデータを用いてビーム幅を変えて認識実験を行い、以下のような結果を得た。

ビーム幅	文認識率	単語認識率
200	59.32 %	76.73%
400	77.97 %	91.51%
800	77.97 %	95.28%
1600	79.66 %	95.60%
full	79.66 %	95.60%

5. まとめ

HMMを認識の基本とし、有限状態オートマトンで表現された正規文法からなるネットワークに、Viterbiサーチを行なう連続音声認識を情報検索システムの検索文、及びコマンドの入力に適用した。

参考文献

- 1) 北原他、本予稿集 (平成3-3)
- 2) Ney, H. et al: "A Data Driven Organization of the Dynamic Programming Beam Search for Continuous Speech Recognition" JEEE ICASSP Apr. 1987